REPORT R-368 OCTOBER, 1967

COORDINATED SCIENCE LABORATORY

A GEOMETRIC APPROACH TO CODING THEORY WITH APPLICATION TO INFORMATION RETRIEVAL

DAVID T.. CHOW

UNIVERSITY OF ILLINOIS - URBANA, ILLINOIS

89

This work was suppor:ed in part by the Joint Services Electronics Program (U.S. Army, U.S. Navy, and U.S. Air Force) under Contra:t DAAB-07-67-C-0199;and in part by the National Science Foundation Grant GK-690.

ACKNOWLEDGMENT

The author wishes to express his deepest gratitude to his advisor Professor R. T. Chien for his inspiring guidance, encouragement, advice, and suggestions throughout the period of research for this thesis.

He would like to thank Professor F. P. Preparata, Professor Franz Hohn and his colleague, Dr. Vincent Lum, for many helpful discussions. The accurate and speedy preparation of the manuscript by Mrs. H. Corray is greatly appreciated.

The support rendered by the Coordinated Science Laboratory is gratefully acknowledged. This research was supported in part by the National Science Foundation Grant GK-690 and in part by the Joint Services Electronic Program (U. S. Army, U.S. Navy, and U.S. Air Force) under Contract No. DAAB-07-67-C-0199.

TABLE OF CONTENTS

Errata:

Page 1, line 19: Change "to described" to "to describe".

Page 1, last line: Change "cycles" to "cyclic".

Page 2, line 20: Change "are" to "is".

Page 4, second line after (2.2): Add "to" after "corresponds".

Page 4, line 4 from the bottom: Change "An" to "A".

Page 5, line 20: Change two "an"'s to two "a"'s.

Page 5, line 21: Change "ambiquity" to "ambiguity".

Page 6, line 17: Change "is" to "are".

Page 7, line 6: Change "an" to "a".

Page 8, line 23: Change "is" to "are".

Page 15, line 1: Delete "of" before "G".

Page 15, line 5: Delete "of".

Page 15, line 6: Change "an" to "a".

Page 16, line 24: Change "was" to "has".

Page 19, line 3 from bottom: Change "an" to "a".

Page 19, last line: Change "an" to "a".

Page 21, line 8: Change "lineary" to "linear".

Page 24, line 2 from the bottom: Change "an" to "a".

Page 25, last line: Change "space" to "spaces".

Page 26, line 2: Change "with" to "will".

Page 28, lines 4 and 5: Change "p- presentation" to "p-ary representation".

Page 29, line 6: Change "In next" to "Next".

Page 40, line 16: Change "invarient" to "invariant".

Page 45, line 4 from bottom: Change "balance" to "balanced".

Page 48, line 4 after table 4.1: Change "an" to "a".

Page 52, last line: Change "is the" to "in the".

Page 54, Table 4.3, upper left side: Change "d_u" to "d".

Page 55, line 15: Change "In other to" to "In order to".

Page 55, line 16: Change "process" to "processing".

Page 57, line 1: Change "cost" to "coset".

Page 57, line 4 from bottom: Change "code of constant weight codes" to "codes of constant weight".

Page 59, first line: Change "(5.4)" to "(5.1)".

Page 59, line 7: Change "$v^{(i)}$" to "$v_u^{(i)}$".

Page 60, line 4 from bottom: Change "superimposed" to "zero-false-drop".

Page 62, line 3: Change "elemently symmetrical" to "elementary symmetric".

Page 62, line 6: Change "$(-1)^{wd}$" to "$(-x)^{wd}$".

Page 62, line 9: Change "hardward" to "hardware".

Page 69, line 15: Change "odded" to "odd".

Page 69, line 23: Change "isomorphis" to "isomorphic".

Page 79, line 2 in reference 8: Insert "Science Laboratory" after "Coordinated".

Page 79, line 3 in reference 12: Change "Balakrishuan" to "Balakrishnan".

Page 80, line 1 in reference 19: Change "Zieler" to "Zierler".

Page 80: Add "Sept. 1960" to the last line.

I. INTRODUCTION

Codes for correcting large multiple random errors are not used extensively in practical data transmission systems because of equipment complexity. Threshold decoding is a method of error correction which is especially suitable for machine implementation because the logical circuit to realize the threshold decoding is usually very simple. Finding cyclic codes that can be decoded by threshold logic becomes important.

Threshold decoding of block codes was introduced by Reed who devised a decoding scheme for the class of codes discovered by Muller [14]. Massey devised many threshold decoding algorithms for recurrent codes as well as block codes. His book [13] "Threshold decoding" includes a comprehensive discussion of the work on threshold decoding for block codes before 1963. Rudolph's [16] threshold decoding algorithm differs from previous algorithms in that the estimates (parity checks) are not necessarily orthogonal. His projective geometry codes are specified in terms of parity check matrices. The determination of the number of check digits lies on the determination of the rank of the parity check matrix which is often not easy, especially when the code length becomes large. It is therefore necessary to develop a theory to described his code in terms of roots of the generator polynomial. The generator polynomial can then be obtained by multiplying irreducible polynomials found from a mathematical table such as the one from reference [14]. Description of the code in term of the generator polynomial is essential in simplifying implementation.

Weldon [17] has discoved a class of cyclic codes based on difference sets. Graham and MacWilliams [4] have studied the number of information digits of difference-set cyclic codes. The class of difference-set cyclic codes is a subclass of Rudolph's projective geometry codes. Kasami [6] has shown that Reed-Muller codes are equivalent to primitive cycles codes with an overall parity check bit added. The cyclic property of these codes simplifies the decoding algorithm of the Reed-Muller codes and also makes them more tractable mathematically. These codes can now be described in terms of the roots of the generator polynomials which suggests a natural generalization to non-binary cases. Weldon has investigated non-primitive Reed-Muller codes [18] which include as subclasses the primitive Reed-Muller codes and difference-set codes and has found a decoding algorithm for them. The non-primitive Reed-Muller code is a subcode of the Rudolph's projective geometry code, and the decoding algorithm for the non-primitive Reed-Muller code is applicable to the Rudolph's projective geometry code. In this thesis, two related classes of codes derived from Euclidean geometries are presented. We call them Euclidean geometry codes and modified Euclidean geometry codes. The generator polynomial of a Euclidean geometry code is (x - 1) times that of the corresponding modified Euclidean geometry code. The code symbols of these codes can be chosen from any field containing a particular prime field GF (p). The dual of a Euclidean geometry code over GF (q) is a subcode of a q-ary (q is a power of prime p) modified Reed-Muller code which contains the parity checks required to make majority voting. We derive a class of codes from projective geometries in terms of the roots of generator polynomials. The discovery of these codes are independent of Weldon's work on non-primitive Reed-Muller code [18] . These codes are better than the corresponding non-primitive Reed-Muller codes in general because they have more information digits and have the same error-correcting ability by L-step orthogonalization procedure. Theoretically Rudolph's projective geometry codes contain the newly established projective geometry codes as subcodes. So far we have not found any case in which they are different. The codes from finite geometries including Euclidean geometries and projective geometries are subcodes of BCH codes of the same length. Thus these codes are not as efficient as BCH codes in general. However, for the most interesting values of code length and rate the difference between finite geometry codes and BCH codes is slight. In an attempt to find a general threshold decoding algorithm for BCH codes, we found that a class of BCH codes cannot be L-step orthogonalized. However, all codes (including some BCH codes) whose extension codes are invariant under a doubly transitive permutation group can be decoded by one-step threshold decoding. Some of the BCH codes turned out to be comparable with codes related to finite geometries by this method.

Recently, coding has been applied to information retrieval. Kautz and Singleton [9] have proposed using zero-false-drop codes for information retrieval. Chien and Frazer [3] have derived methods for document retrieval from algebraic coding theory. Two new classes of zero-false-drop codes have been derived from finite geometries. These codes provide more useful parameters than the previous ones. Investigation has been made of the use of error-correcting codes for information retrieval. Several interesting results have been obtained.

The material in this paper is arranged as follows. In chapter 2, we present the codes derived from Euclidean geometries. In chapter 3, the polynomial version of projective geometry codes is given. In chapter 4 we investigate the threshold decoding of cyclic codes, including BCH codes and finite geometry codes. In chapter 5, the application of coding theory to information retrieval is presented. Finally we have, in chapter 6, the conclusion and suggestions for promising areas of future research.

II. CODES DERIVED FROM EUCLIDEAN GEOMETRIES

2.1 Preliminaries

Codes derived from Euclidean geometries are closely related to Reed-Muller codes, we first introduce some background concerning Euclidean geometries and then indicate the connection between Euclidean geometries and Reed-Muller codes.

Let α be a primitive element in $GF(q^m)$. As the elements $\alpha^i (i = 0, 1 \ldots\ldots, m-1)$ are linearly independent over $GF(q)$, we may write

$$\alpha^j = \sum_{i=0}^{m-1} v_{ij}\alpha^i \quad ; \quad 0 \leq j \leq q^m - 2 \tag{2.1}$$

where v_{ij} is in $GF(q)$.

Let G be a matrix defined as

$$G = \begin{bmatrix} 0 & v_{(m-1)0} & v_{(m-1)1} & \cdot\ \cdot & v_{(m-1)(q^m-2)} \\ & \cdot\ \cdot & \cdot & \cdot & \cdot \\ 0 & v_{10} & v_{11} & \cdot\ \cdot & v_{1(q^m-2)} \\ 0 & v_{00} & v_{01} & \cdot\ \cdot & v_{0(q^m-2)} \end{bmatrix} \tag{2.2}$$

The first column of G corresponds to the zero element in $GF(q^m)$ and the j^{th} column $(2 \leq j \leq q^m)$ corresponds α^{j-2} in $GF(q^m)$.

We can associate each column of the matrix in equation (2.2) a point in a Euclidean geometry of m-dimension over $GF(q)$, denoted as $EG(m,q)$ [2]. $EG(m,q)$ consists of q^m points $0, 1, \alpha, \alpha^2, \ldots, \alpha^{q^m-2}$. An u-dimensional flat of $EG(m,q)$ consists of q^u points $a_1\alpha_1 + a_2\alpha_2 + \ldots + a_u\alpha_u + \gamma$ where α_i $(1 \leq i \leq u)$ are elements in $GF(q^m)$ and are linearly independent over $GF(q)$. $a_1, a_2, \ldots\ldots, a_u$ may run independently over GF(q) and γ is a fixed element in GF(q^m) [2]. These q^u points are solutions to the m-u linearly independent equations of m unknowns. A vector associated with an u-dimensional flat is defined to be a vector with 1 (multiplicative identity in GF(q)) in the positions corresponding to all points in this flat and 0 (additive identity in GF(q))otherwise.

Kasami et al. [6] has defined a q-ary ν-th order modified Reed-Muller code to be a cyclic code whose generator polynomial contains the roots α^h for all positive integers h such that the weight of h over base q or the "digit" sum over the real field of q-ary representation of h is greater than zero but less than m(q-1) - ν. That is, if

$$h = \sum_{i=0}^{m-1} h_i q^i \quad ; \quad 0 \leq h_i \leq q - 1$$

then $$w_q(h) = \sum_{i=0}^{m-1} h_i .$$ α^h is a root of a q-ary modified Reed-Muller code provided

$$0 < w_q(h) = \sum_{i=0}^{m-1} h_i < m(q-1) - \nu \qquad (2.3)$$

It has been shown [6] that a q-ary ν-th order Reed-Muller code can be obtained by adding an overall parity check as a first digit to every code word of a q-ary ν-th order modified Reed-Muller code. It is well known [14] that a binary (m-u)-th order Reed-Muller code contains all vectors associated with u-dimensional flats. From now on, we shall use the term "an u-dimensional flat" instead of "the vector associated with an u-dimensional flat" when no ambiquity arises. It has been shown [6] that a q-ary (m-u) (q-1)-th order Reed-Muller code contains all u-dimensional flats of

EG(m, q) passing through the point corresponding to the zero element in $GF(q^m)$. In general, a q-ary (m-u) (q-1)-th order Reed-Muller code contains all u-dimensional flats of EG(m, q). The proof can be given by an argument similar to that used in reference [6].

2.2 Euclidean Geometry Codes

Let q be equal to p^s where p is a prime and s is a positive integer. In this section, we are going to present a cyclic code over GF(p) which has as parity checks all u-dimensional flats of EG(m, q) with their first digits deleted.

From the preceding section, q-ary (m-u) (q-1)-th order modified Reed-Muller code contains all the vectors v_u obtained by deleting the first digits of the vectors associated with the u-dimensional flats. From equation (2.3),the generator polynomial $g_1(x)$ of a q-ary (m-u) (q-1)-th order modified Reed-Muller code contains α^h as roots for h satisfying

$$0 < w_q(h) < u\,(q-1) \tag{2.4}$$

Let $v_u(x)$ be the polynomial corresponding to v_u. $v_u(x)$ is a polynomial whose coefficients is either 1 or 0 hence $v_u(x)$ can be considered as a polynomial over GF(p). Since $v_u(x^p)$ is equal to $(v_u(x))^p$, $v_u(x)$ contains α^{hp} as a root if it contains α^h as a root.

Let $C_o^{(q)}$ be a code over GF(q) whose generator polynomial $g_2(x)$ contains α^{hp^j} as roots for h satisfying the condition (2.4). $C_o^{(q)}$ is a subcode of a q-ary (m-u) (q-1)-th order modified Reed-Muller Code and $C_o^{(q)}$ contains all u-dimensional flats with their first digits deleted.

$g_2(x)$ is a polynomial over GF(p). We now show that the cyclic code C_0 over GF(p) with $g_2(x)$ as its generator polynomial also contains the u-dimensional flats with their first digits deleted as code words. $g_2(x)$ is the least common multiple of the minimal polynomials $m_i(x)$ over GF(p) of the roots α^h for h satisfying the condition (2.4). The polynomial $v_u(x)$ associated with an u-dimensional flat is divisible by any such $m_i(x)$, hence $v_u(x)$ is divisible by the least common multiple of these $m_i(x)$. $v_u(x)$ has its coefficients in GF(p) and is divisible by $g_2(x)$. The code C_0 contains all u-dimensional flats with their first digits deleted as code words.

Let C be the dual of the code C_0. $g_2(x)$ contains α^{hp^j} as roots for h satisfying the condition (2.4). The reciprocal polynomial $g_2^*(x)$ of $g_2(x)$ contains α^{hp^j} as roots for h satisfying

$$m(q-1) > w_q(h) > (m-u)(q-1) \qquad (2.5)$$

The generator polynomial $g_e(x)$ of the code C contains all q^m-1-th roots of unity which are not roots of $g_2^*(x)$. α^h are roots of $g_e(x)$ for any nonnegative integer h less than q^m-1 and satisfying the condition

$$0 \leq w_q(hp^j) \leq (m-u)(q-1) \quad ; \quad 0 \leq j \leq s-1 \qquad (2.6)$$

Next we show that the lower bound of the minimum distance of the code C is

$$d = 2 + q + \ldots. + q^{m-u} \qquad (2.7)$$

This can be achieved by showing that any nonnegative integer h less than $1 + q + \ldots + q^{m-u}$ satisfies the condition (2.6). Let

$$h = \sum_{i=0}^{m-1} h_i q^i \quad ; 0 \leq h_i \leq q-1 \qquad (2.8)$$

If $h_i \neq 0$ for all i such that $0 \leq i \leq m-u$, then $h \geq 1+q+\ldots+q^{m-u}$. Thus for any nonnegative integer h which is less than $1+q+\ldots+q^{m-u}$,

$$h_i = 0 \quad \text{for } m-u+1 \leq i \leq m-1 \text{ and} \tag{2.9}$$

$$\text{for at least one } i \text{ in the range of } 0 \leq i \leq m-u$$

For any h satisfying condition (2.9), one can easily verify that $w_q(hp^j) \leq (m-u)(q-1)$ for all j. Thus the generator polynomial of the code C contains $1+q+\ldots+q^{m-u}$ consecutive roots. The minimum distance of this code is at least $2+q+\ldots..+q^{m-u}$.

Since C and C_0 are dual codes, all code words of C satisfy the parity checks specified by u-dimensional flats.

We shall call the code C a Euclidean geometry code with parameters m, u, q. The code exists for any power of prime $q(q=p^s$, where p is a prime and s is a positive integer), any positive integer m and any integer u fo. $1 \leq u \leq m-1$.

Theorem 2.1: A Euclidean geometry code with parameter m, u, q has code length q^m-1. The number of pariiy checks of this code is the number of nonnegative integer h less than q^m-1 such that the weight of the integers hp^j ($0 \leq j \leq s-1$) over base q is no more than $(m-u)(q-1)$. The minimum distance of this code is at least $2+q+\ldots+q^{m-u}$.

The mcst important subclass of Euclidean geometry codes is binary codes. That is, the case when p equals to 2. We list some binary Euclidean Geometry codes in Table 2.1. The entries in this table is (n, k, d) where n is the code length, k is the number of information digits and d is the lower bound on minimum distance ($d = 2+q+\ldots.+q^{m-u}$).

Table 2.1 Binary Euclidean Geometry Cyclic Codes

EG(m,q)	u = m-1	u = m-2	u = m-3	u = m-4
EG (3,2)	(7,3,4)			
EG (4,2)	(15,10,4)	(15,4,8)		
EG (5,2)	(31,25,4)	(31,15,8)	(31,5,16)	
EG (6,2)	(63,56,4)	(63,41,8)	(63,21,16)	(63,6,32)
EG (2,4)	(15,6,6)			
EG (3,4)	(63,47,6)	(63,12,22)		
EG (4,4)	(255,230,6)	(255,126,22)	(255,20,86)	
EG (5,4)	(1023,987,6)	(1023,747,22)	(1023,287,86)	(1023,32,342)
EG (2,8)	(63,36,10)			
EG (3,8)	(511,447,10)	(511,138,74)		
EG (4,8)	(4095,3970,10)	(4095,2584,74)	(4095,405,586)	
EG (2,16)	(255,174,18)			
EG (3,16)	(4095,3839,38)	(4095,1376,274)		
EG (2,32)	(1023,780,34)			
EG (2,64)	(4095,3366,66)			

The first four rows of this Table are codes associated with binary Reed-Muller codes and are known. (15,6) code is a BCH code. The rest of the codes seem to be new.

2.3 Decoding of Euclidean Geometry Codes

In this section, we show that by using u-step orthogonalization procedure similar to the Reed decoding algorithm we can decode the Euclidean geometry code C with parameters m, u, q to the bound on the minimun distance d [18].

The code C satisfies the u-dimensional flat parity checks. In the first step, we determine u-1-dimensional flat check sums from u-dimensional flats. In the u'-th step ($1 \leq u' \leq u$) we determine u-u'-dimensional flat check sums from u-u'+ 1-dimensional flats.

The following theorem is essential to the implementation of the decoder.

Theorem 2.2 : For a given u'-1-dimensional flat ($1 \leq u' \leq u$), the number of u'-dimensional flats containing this u'-1-dimensional flat is $1 + q + \dots + q^{m-u'}$. Any two of these u'-dimensional flats has no points in common except the points in this u'-1-dimensional flat.

Proof: Consider a particular u'-1-dimensional flat which consists of $q^{u'-1}$ points $a_1\alpha_1 + a_2\alpha_2 + \dots + a_{u'-1}\alpha_{u'-1}$. $\alpha_i (1 \leq i \leq u'-1)$ are linearly independent points, a_i $(1 \leq i \leq u'-1)$ may run independently over GF(q). The u'-dimensional flat containing this u'-1-dimensional flat consists of the points $a_1\alpha_1 + a_2\alpha_2 + \dots + a_{u'-1}\alpha_{u'-1} + a_{u'}\alpha_{u'}$. $\alpha_i (1 \leq i \leq u')$ are linearly independent points. $a_{u'}$ runs over all elements in GF(q). The number of choice of $\alpha_{u'}$ is $q^m - q^{u'-1}$. In a fixed u'-dimensional flat, the number of choice of $\alpha_{u'}$ is $q^{u'} - q^{u'-1}$. The number of distinct u -dimensional flats containing this u'-1-dimensional flat is $(q^m - q^{u'-1}) / (q^{u'} - q^{u'-1})$ or $1 + q + \dots + q^{m-u'}$. The u'-1-dimensional flat and a point not in this u'-1-dimensional flat specify a u'-dimensional flat uniquely. If two u'-dimensional flats have a point outside this u'-1 dimensional flat in common, they must be identical.

In general, the u'-1-dimensional flat consists of points of the form $a_1\alpha_1 + a_2\alpha_2 + \ldots + a_{u'-1}\alpha_{u'-1} + \gamma$ where γ is a fixed element in $GF(q^m)$. The u'-dimensional flat containing this u'-1-dimensional flat consists of points $a_1\alpha_1 + a_2\alpha_2 + \ldots + a_{u'-1}\alpha_{u'-1} + a_{u'}\alpha_{u'} + \gamma$. Adding γ to the points in EG(m, q) can be considered as a permutation of points in EG(m, q). The argument in the preceding paragraph is still true for this u'-1-dimensional flat. This proves the theorem.

The number $1 + q + \ldots + q^{m-u'}$ is no less than $1 + q + \ldots + q^{m-u}$ for any u' $(1 \leq u' \leq u)$. We can always pick $d-1$ $(d = 2 + q + \ldots + q^{m-u})$ number of u'-dimensional flats orthogonal on a particular u'-1-dimensional flat to determine a parity check sum corresponding to this u'-1-dimensional flat. The determination will be correct provided the number of errors occurred is no more than $\left[(d-1)/2\right]$. The decoder consists of u levels of majority logic. In the u-th level, we need a majority gate to determine the point (0-dimensional flat) corresponding the first digit position of a code word. The input to this majority gate is $1 + q + \ldots + q^{m-u}$ 1-dimensional flat parity check sums orthogonal to this point. We use $1 + q + \ldots . + q^{m-u}$ majority gates in the $(u - 1)$-th level. The decoder is tree-like. In the j-th level $(1 \leq j \leq u)$, we use $(1 + q + \ldots + q^{m-u})^{u-j}$ majority gates to determine the same number of (u-j)-dimensional flats. The total number of majority gates is

$$I = \sum_{j=1}^{u} (1 + q + \ldots . + q^{m-u})^{j-1}$$

The choice of (u-j)-dimensional flat parity check sums in j-th level is not unique. The construction of the decoder is not unique. Furthermore, the u'$(1 \leq u' \leq u)$ dimensional flat used in the majority voting may not be linearly independent. We may not need I number of majority gates if this is the case. The tree-like decoder is not necessarily the best one. Simplification in circuitry is possible by detail evaluation of the dependency of the parity checks required.

We now give an example of the binary (15, 6) Euclidean Geometry code to illustrate the method of obtaining parity checks required for threshold decoding.

Example: Take $q = 2^2$, $m = 2$, and $u = 1$. Let α be a root of primitive polynomial $x^4 + x + 1$ over $GF(2^4)$. α is a primitive element of $GF(2^4)$.

All the elements in $GF(2^4)$ are linear combinations over $GF(2)$ of $1, \alpha, \alpha^2, \alpha^3$ as follows:

$$\begin{bmatrix} 0 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 & \alpha^8 & \alpha^9 & \alpha^{10} & \alpha^{11} & \alpha^{12} & \alpha^{13} & \alpha^{14} \end{bmatrix}$$
$$= \begin{bmatrix} \alpha^3 & \alpha^2 & \alpha & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \qquad (2.10)$$

Let $\beta = \alpha^5$, then β is a primitive element of $GF(2^2)$. We can consider $GF(2^4)$ as an extension field of $GF(2^2)$. α is a primitive element of $GF(2^4)$. All elements in $GF(2^4)$ are linear combinations over $GF(2^2)$ of $1, \alpha$. All elements in $GF(2^2)$ can be written as linear combinations of $1, \beta$. From equation (2.10), all elements in $GF(2^4)$ can be written as linear combinations of $1, \beta, \alpha, \alpha\beta$ over $GF(2)$ as follows:

$$\begin{bmatrix} 0 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 & \alpha^8 & \alpha^9 & \alpha^{10} & \alpha^{11} & \alpha^{12} & \alpha^{13} & \alpha^{14} \end{bmatrix}$$
$$= \begin{bmatrix} \alpha\beta & \alpha & \beta & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} \qquad (2.11)$$

Since β satisfies the primitive polynomial $x^2 + x + 1$ over GF(2),

$$\beta^2 = \beta + 1 \qquad (2.12)$$

Equation (2.11) can be rewritten as

$$\begin{bmatrix} 0 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 & \alpha^8 & \alpha^9 & \alpha^{10} & \alpha^{11} & \alpha^{12} & \alpha^{13} & \alpha^{14} \end{bmatrix}$$
$$= \begin{bmatrix} \alpha & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 1 & \beta^2 & 1 & 0 & \beta & \beta & 1 & \beta & 0 & \beta^2 & \beta^2 & \beta & \beta^2 \\ 0 & 1 & 0 & \beta & \beta & 1 & \beta & 0 & \beta^2 & \beta^2 & \beta & \beta^2 & 0 & 1 & 1 & \beta^2 \end{bmatrix} \qquad (2.13)$$

It is easy to see from equation (2.13) that α is a root of a primitive polynomial $x^2 + x + \beta$ over $GF(2^2)$.

The generator polynomial $g_e(x)$ of the Euclidean geometry code C contains α^h as roots for the integers h such that

$$0 \leq w_4(h\,2^j) \leq (2-1)(4-1) \quad ; \quad 0 \leq j \leq 2-1$$

The integers h satisfy this condition are 0, 1, 2, 4, 8, 3, 6, 9 and 12.

The generator polynomial $g_2(x)$ of the dual C_0 of the code C contains α^h as roots for the h's equal to 1, 2, 4, 8, 5 and 10. The q-ary (q=4) third order modified Reed-Muller code has generator polynomial $g_1(x)$ contains α^h as roots for the h's equal to 1, 2, 4, 8 and 5. The code $C_0^{(q)}$ over

$GF(2^2)$ with $g_2(x)$ as its generator polynomial is a proper subcode of the q-ary third order modified Reed-Muller code.

From matrix (2.13), the 1-dimensional flats over $GF(2^2)$ of $EG(2,2^2)$ passing through the point $\alpha^0 = 1$ are as follows.

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad (2.14)$$

The i-th row ($1 \leq i \leq 5$) corresponding to a 1-dimensional flat consists of $a\alpha^{i-1} + 1$ where a runs over $GF(2^2)$.

By deleting the first column of matrix (2.14), we have a set of 5 parity checks orthogonal to the point α^0. These are parity checks for the binary (15, 6) code.

2.4 Modified Euclidean Geometry Codes

In this section, we define a modified Euclidean geometry code which has one more information digit than the corresponding Euclidean geometry code and has a lower bound for minimum distance less than that of the latter code by one and can be u-step decoded up to this bound.

Let $g_e(x)$ be the generator polynomial of the Euclidean geometry code with parameters m, u, q. We define the corresponding modified Euclidean geometry code by the code generated by $g_e(x)/(x-1)$. Obviously, the lower bound for the minimum distance of this code is $1 + q + \ldots. + q^{m-u}$. q-ary (m-u) (q-1)-th order Reed-Muller code contains all the vectors associated with u-dimensional flats. The generator matrix of G of this Reed-Muller code would have the property that the vector with all 1 entries is the only vector with nonzero entry in the first digit. If we delete the all 1 vector in G, the row space of the new matrix contains all the vectors in the row space of whose first digit is equal to zero. Let v_u' be a vector obtained by deleting the first digit of a vector corresponding to an u-dimensional flat which does not pass through the first point. From this argument, a code with generator polynomial (x-1) $g_1(x)$ contains all the vectors v_u' where $g_1(x)$ is the generator polynomial of q-ary (m-u) (q-1)-th order modified Reed-Muller code. Thus the modified Euclidean geometry code satisfies all u-dimensional flat parity checks which do not pass through the deleted point. The total number of u'-dimensional flats $(1 \leq u' \leq u)$ containing a particular (u'-1)-dimensional flat which does not pass through the deleted point is $1 + q + ... + q^{m-u'}$. Only one of these u'-dimensional flat contains the deleted point. We can decode this modified Euclidean geometry code to the distance $1 + q + ... + q^{m-u}$ by u-step orthogonalization procedure.

We can obtain some binary modified Euclidean geometry codes easily from Table 2. 1.

III. CODES DERIVED FROM PROJECTIVE GEOMETRIES

3.1 Introduction

A class of cyclic codes suitable for threshold decoding has been developed by Rudolph [16] by using the properties of projective geometries. He first found a majority decoding algorithm which does not require the parity checks used in majority voting to be orthogonal. The guaranteed error correction of a code can be determined easily as the parity check matrix is an incidence matrix of a balanced incomplete block design which is also cyclic. However, it is often necessary to compute the rank of the parity check matrix individually. Weldon [18] defined and developed the non-primitive Reed-Muller codes in terms of the roots of generator polynomials. An important subclass of non-primitive Reed-Muller codes are subcodes of Rudolph's projective geometry codes. Following Weldon's approach, the generator polynomials of these non-primitive Reed-Muller codes can be found easily and a more powerful decoding scheme is also applicable. However, the Rudolph version of these codes generally possess a larger number of information digits.

In this chapter, we describe codes derived from projective geometries in terms of the roots of the generator polynomials. These codes are better than Weldon's non-primitive Reed-Muller codes because they have more information digits in general. These codes are generally subcodes of Rudolph's codes. So far we have not yet found any of the cases that these codes are not identical to Rudolph's projective geometry codes. A better code which contains the new code as a subcode and which was all required parity checks is given. In some special case, this code is identical to the Rudolph's projective geometry code. The description of the generator polynomial of this code is somewhat less easy than the previous one.

In an attempt to find the polynomial version of the Rudolph's code, we have succeeded, independently of Weldon's work, in constructing the generator polynomials of a class of cyclic codes. It is shown in section 3.3 that these codes include Weldon's codes as subcodes in general and they possess a larger number of information digits in a number of cases.

3.2 Rudolph's Projective Geometry Codes

First, let us introduce Rudolph's majority decoding algorithm. Let $A = \left[a_{ij}\right]$, $i = 0, 1, \ldots, b-1$; denote the parity check matrix of a cyclic code over GF(p). Suppose the leftmost column of A contains r nonzero elements, namely, $a_{i_k 0}$, $k = 1, 2, \ldots, r$. Consider $A_o = \left[a_{i_k j}\right]$, $k = 1, 2, \ldots, r$ a submatrix of A. A received sequence $B = (b_o, b_1, \ldots b_{v-1})$ is a vector sum of a transmitted code word $C = (c_o, c_1, \ldots, c_{v-1})$ and an error vector $E = (e_o, e_1, \ldots, e_{v-1})$. To decode received digit b_o, we first multiply the matrix A_o by the transpose of the received sequence B and set the product A_oB^T equal to zero. The resulting equations are

$$\sum_{j=0}^{v-1} a_{i_k j}\, b_j = 0 \qquad k = 1, 2, \ldots, r \tag{3.1}$$

Treating b_o as an unknown and solving

$$b_o = - a_{i_k o}^{-1} \sum_{j=1}^{v-1} a_{i_k j}\, b_j \qquad k = 1, 2, \ldots, r \tag{3.2}$$

Denote the r "estimates" of the first received digit by $b_o^{(k)}$, $k = 1, 2, \ldots, r$. One additional estimator is the identity $b_o^{(0)} = b_o$. Now set the decoded symbol $\hat{c}_o$ equal to that value of GF(p) assumed by the largest fraction of the r + 1 estimates $b_o^{(k)}$. For a cyclic code this scheme for decoding the first digit also decodes the other v -1 digits.

In a balanced incomplete block design, we have v objects arranged in b blocks. Each block contains k_1 distinct objects. Each object occurs r times and each pair of objects occurs together in λ times. Block design is conveniently represented by a b by v incidence matrix $S = \left[s_{ij} \right]$ where $s_{ij} = 1$ if i-th block contains j-th element, $s_{ij} = 0$ otherwise. Elemently conditions for the existence of a (v, k_1, r, b, λ) balanced incomplete block design are as follows.

1. $vr = b\, k_1$

2. $\lambda\,(v-1) = r(k_1 - 1)$ (3.3)

If the parity check matrix A is the incidence matrix S, then the r by v submatrix will have r 1's in its leftmost column and λ 1's in all other columns. This leads to a set of $r + 1$ estimators (including the identity with each b_j appearing in no more than λ equations. The decoding algorithm is capable of correcting any combination of e or fewer errors where $e = \left[r/2\lambda \right]$. The brackets denote "integer part of."

A balanced incomplete block design is called cyclic if every cyclic permutation of a row of the incidence matrix A is also a row of A. One well-known class of cyclic designs is associated with projective geometries.

Denote by $PG(m_1, q)$ the projective geometry of dimension m_1 over $GF(q)$. For each $u(1 \leq u < m_1)$, one can associate a cyclic incidence matrix A such that the columns correspond to the points and the rows correspond to all possible u-spaces of $PG(m_1, q)$. The error-correcting ability is

$$e = \left[r/2\lambda \right] = \left[\frac{q^{m_1} - 1}{2(q^u - 1)} \right] \qquad (3.4)$$

The number of check digits however is not known explicitly. A computational procedure is of course possible by determining the rank of A.

3.3 A Method for Determining the Generator Polynomials of Projective Geometry Codes

Rudolph describes his projective geometry codes through the parity check matrices. In this section a procedure is described for finding the generator polynomials of a new class of cyclic codes. It is shown that the code polynomials of the codes generated do satisfy Rudolph's parity check equations. Hence the new codes are subcodes of the codes specified by Rudolph.

Let $q = p^s$. Let α be a primitive element of $GF(q^m)$. Let n be $(q^m-1) / (q-1)$. As the elements $\alpha^i (1 \leq i \leq m_1,\ m_1 = m-1)$ are linearly independent over $GF(q)$, we may write

$$\alpha^j = \sum_{i=1}^{m_1} v_{ij}\, \alpha^i \qquad 0 \leq j \leq q^m-2 \qquad (3.5)$$

where v_{ij} is in $GF(q)$. Arrange the coefficients in matrix form and define

$$G = \begin{bmatrix} u_{m_1} \\ \cdot \\ u_1 \\ u_o \end{bmatrix} = \begin{bmatrix} v_{m_1 0} & v_{m_1 1} & \cdot\ \cdot & v_{m_1(n-1)} \\ \cdot & \cdot & \cdot\ \cdot & \\ v_{10} & v_{11} & \cdot\ \cdot & v_{1(n-1)} \\ v_{00} & v_{01} & \cdot\ \cdot & v_{0(n-1)} \end{bmatrix} \qquad (3.6)$$

We can associate each colum. ,corresponding to α^i) of this matrix to a point of $PG(m_1,q)$ [1] .

An u-space of $PG(m_1,q)$ is defined to be the totality of the points linearly dependent upon a set of $u + 1$ linearly independent points. A vector associated with an u-space is a vector such that its component is equal to 1 if the position of this component corresponds to a point contained in this u-space and the component is equal to 0 otherwise.

We first show that the vectors associated with u-spaces can be expressed as linear combinations of vectors related to the product of the powers of the vectors $u_o, u_1, \ldots, u_{m_1}$. Let

$$\begin{aligned} u_i u_j &= (v_{i0}, v_{i1}, \ldots, v_{i(n-1)})\ (v_{j0}, v_{j1}, \ldots, v_{j(n-1)}) \\ &= (v_{i0}v_{j0}, v_{i1}v_{j1}, \ldots, v_{i(n-1)}v_{j(n-1)}), \quad 0 \le i,j \le m_1 \end{aligned} \tag{3.7}$$

where $v_{ik}v_{jk}$ is a product of v_{ik} and v_{jk} in GF(q). For a positive integer ℓ_i, define $u_i^{\ell_i}$ to be u_i multiplied by itself ℓ_i times. Define

$$u_I = u_0^o\, u_1^o \ldots\ldots u_{m_1}^o = (1, 1, \ldots, 1). \tag{3.8}$$

Let L_u be a matrix

$$L_u = \begin{bmatrix} u_0^{\ell_o}\ u_1^{\ell_1} \ldots,\ u_{m_1}^{\ell_{m_1}} \end{bmatrix} \tag{3.9}$$

whose rows are the vectors $u_0^{\ell_o}\, u_1^{\ell_1} \ldots\ldots u_{m_1}^{\ell_{m_1}}$ with $\ell_o, \ell_1, \ldots, \ell_{m_1}$ satisfying the condition

$$\sum_{i=0}^{m_1} \ell_i = c(q-1)\ ; \quad 0 \le c \le (m_1-u), \quad 0 \le \ell_i \le q-1 \tag{3.10}$$

The u-space which contains $\alpha^o, \alpha^1, \ldots\ldots, \alpha^u$, corresponds to the vector

$$\frac{1}{(-1)^{m_1-u}} \prod_{i=u+1}^{m_1} (u_i^{q-1} - u_I) \tag{3.11}$$

This vector is in the row space of L_u. Consider any u-space containing $\alpha^{d_0}, \alpha^{d_1}, \ldots, \alpha^{d_u}$ linearly independent points over GF(q), then there exists a nonsingular matrix M over GF(q) such that

$$\left[\alpha^{0}, \alpha^{1}, \ldots, \alpha^{u}\right] = M\left[\alpha^{d_0}, \alpha^{d_1}, \ldots, \alpha^{d_u}\right] \tag{3.12}$$

Let $\left[x_{m_1}, \ldots\ldots, x_1, x_0\right]^T = M\left[u_{m_1}, \ldots, u_1, u_o\right]^T$. The vector

$$\frac{1}{(-1)^{m_1-u}} \prod_{i=u+1}^{m_1} (x_i^{q-1} - u_I)$$ corresponds to the u-space

containing $d_o, d_1, \ldots, d_u$. One can easily verify that this vector is a lineary combination of the vectors $u_0^{\ell_o} u_1^{\ell_1} \ldots u_{m_1}^{\ell_{m_1}}$ satisfying the condition (3.10). Thus we have shown the following.

Lemma 3.1 Any vector corresponding to an u-space is in the row space of L_u over GF(q).

$$G_u = \left[1 \;\alpha^h \;\text{----}\; (\alpha^h)^{n-1}\right] \tag{3.13}$$

where h is an integer less than q^m-1 and is a multiple of q-1 and the weight of h over base q is no more than $(m_1 - u)(q-1)$.

We want to show that the row space of G_u over GF(q) is identical to the row space of L_u over GF(q).

Let

$$h = \sum_{i=0}^{m_1} h_i q^i \quad ; \quad 0 \leq h_i \leq q-1 \tag{3.14}$$

$$h = \sum_{i=0}^{m_1} h_i + \sum_{i=0}^{m_1} h_i (q^i - 1) = w_q(h) + \sum_{i=0}^{m_1} h_i(q^i - 1) \tag{3.15}$$

q^i-1 is divisible by q-1. Thus h is divisible by q-1 if and only if $w_q(h)$ is divisible by q-1.

Any integer h in the matrix G_u satisfies

$$w_q(h) = c(q-1) \quad ; \quad 0 \leq c \leq (m_1-u) \tag{3.16}$$

We first show that any row vector of G_u is a linear combination of the rows of L_u. Consider a typical row of matrix (3.13),

$$\left[1 \; \alpha^h \; ---- \; (\alpha^h)^{n-1}\right] \tag{3.17}$$

The matrix consists of m rows. Let

$$(\alpha^h)^j = \sum_{i=0}^{m_1} c_{ij}\alpha^i \tag{3.18}$$

The typical column $(\alpha^h)^j$ is actually as the following.

$$\begin{bmatrix} c_{m_1 j} \\ \cdot \\ \cdot \\ c_{1j} \\ c_{oj} \end{bmatrix} \tag{3.19}$$

We need only to show that c_{ij} $(m_1 \leq i \leq 0)$ is a linear combination over GF(q) of the terms $v_{oj}^{\ell_o} \; v_{1j}^{\ell_1} \cdots v_{m_1 j}^{\ell_{m_1}}$ with $\ell_o, \ell_1, \ldots, \ell_{m_1}$ satisfying condition (3.10) and this linear combination is independent of j.

From equations (3.5) and (3.14), we have

$$(\alpha^h)^j = (\alpha^j)^h = \left(\sum_{i=0}^{m_1} v_{ij}\,\alpha^i\right)^{\sum_{t=0}^{m_1} h_t q^t} \tag{3.20}$$

By expanding equation (3.20)

$$(\alpha^{h})^{j} = \prod_{t=0}^{m_1} \left(\sum_{i=0}^{m_1} v_{ij} \, \alpha^{iq^t} \right)^{h_t}$$

$$= \prod_{t=0}^{m_1} \left(\sum_{0 \le i_1, i_2, \ldots, i_{h_t} \le m_1} v_{i_1 j} v_{i_2 j} \cdots v_{i_{h_t} j} \, \alpha^{(i_1 + i_2 + \ldots + i_{h_t}) q^t} \right) \qquad (3.21)$$

In the t-th factor of equation (3.21), each term in the summation has as coefficient $v_{i_1 j} \; v_{i_2 j} \ldots . \; v_{i_{h_t} j}$ which is a product of h_t of the factors $v_{0j}, v_{1j}, \ldots, v_{m_1 j}$ with repetitions permitted. By expanding equation (3.21)

$$(\alpha^{h})^{j} = \sum_{f} b_{fj} \, \alpha^{f(h)} \qquad (3.22)$$

Where f(h) depends only on h but not on j, b_{fj} has the form

$$b_{fj} = v_{0j}^{\ell'_o} \; v_{1j}^{\ell'_1} \; ---- \; v_{m_1 j}^{\ell'_{m_1}} \qquad (3.23)$$

and

$$\sum_{i=0}^{m_1} \ell'_i = \sum_{t=0}^{m_1} h_t = w_q(h) \qquad (3.24)$$

Let

$$0 \le \ell_i \le q-1 \quad \text{and} \quad \ell_i \equiv \ell'_i \mod q-1$$

then

$$b_{fj} = v_{0j}^{\ell_o} \; v_{1j}^{\ell_1} \; ---- \; v_{m_1 j}^{\ell_{m_1}} \qquad (3.25)$$

with $\ell_o, \ell_1, \ldots, \ell_{m_1}$ satisfying condition (3.10).

$\alpha^{f(h)}$ can be expressed as a linear combination over GF(q) of 1, $\alpha, \ldots, \alpha^{m_1}$. Equation (3.22) can be rewritten as

$$(\alpha^j)^h = \sum_{i=0}^{m_1} c_{ij}\, \alpha^i \tag{3.26}$$

where c_{ij} is a linear combination over GF(q) of b_{fj}. Thus we have shown that the row space of G_u over GF(q) is a subspace of the row space of L_u over GF(q). The dimension of the row space of G_u is equal to the number of h satisfying condition (3.16). The dimension of the row space of L_u over GF(q) is no more than the number of rows of L_u. Let the integer h be corresponding to $(\ell_o, \ell_1, \ldots, \ell_{m_1})$ if $h_i = \ell_i$ for $0 \leq i \leq m_1$. The number of h satisfying condition (3.1o) and the number of $(\ell_o, \ell_1, \ldots, \ell_{m_1})$ satisfying condition (3.10) are equal because there exists one to one correspondence between these h's and these $(\ell_o, \ell_1, \ldots, \ell_{m_1})$'s. The dimension of the row space of G_u is no less than the dimension of the row space of L_u. Hence, we have

Lemma 3.2 The row space of the matrix G_u over GF(q) is equal to the row space of the matrix L_u over GF(q).

Theorem 3.1 Let C be a code with α^h as a root of its generator polynomial g(x) if and only if h is divisible by q-1 and satisfies the following condition.

$$0 \leq w_q(hp^j) \leq (m_1 - u)(q-1)\,; \quad 0 \leq j \leq s-1 \tag{3.27}$$

$w_q(hp^j)$ is a digit sum of the q-ary representation of hp^j. The null space of C contains all u-spaces of $PG(m_1, q)$.

Proof: From lemma 3.1 and lemma 3.2, any vector associated with an u-space is in a code over GF(q) whose generator matrix is equal to G_u. The parity check matrix in this code is

$$H_u = \left[1 \ \alpha^{h'} \ \text{-----} \ (\alpha^{h'})^{n-1} \right] \tag{3.28}$$

where h' is an integer less than $q^m - 1$ and satisfying the condition

$$q-1 \mid h' \qquad 0 < w_q(h') < (u+1)(q-1) \tag{3.29}$$

Let v(x) be a polynomial associated with an u-space. $\alpha^{h'}$ are roots of v(x) for h' satisfying condition (3.29). v(x) has its coefficients over GF(p). $\alpha^{h'}$ is a root of v(x) implies $\alpha^{h'p^j}$ are roots of v(x).

A code whose generator matrix is

$$\left[1 \ \alpha^{h'p^j} \ \text{--------} \ (\alpha^{h'p^j})^{n-1} \right] \tag{3.30}$$

with h' satisfying condition (3.29) satisfies the parity checks associated with all u-spaces of $PG(m_1, q)$.

Thus the generator polynomial of this code has α^h as its roots for h satisfying condition (3.27). This theorem is proved.

BCH bound for this code can be seen as follows. Let $h = t(q-1)$ and $h = \sum_{i=0}^{m_1} h_i q^i$ where $q-1 \geq h_i \geq 0$. For $0 \leq t \leq q^{m_1-u} + q^{m_1-u-1} + \ldots + q$, h_i equal to 0 for $m_1 \geq i \geq (m_1-u+1)$ and not all other h_i's equal to q-1, hence $w_q(t(q-1)) < (m_1-u+1)(q-1)$. One can easily verify that $w_q(t(q-1)p^j) < (m_1-u+1)(q-1)$ for all these t's. Thus the code C defined in theorem 3.1 contains $q^{m_1-u} + q^{m_1-u-1} + \ldots + 1$ consecutive roots. Weldon's decoding algorithm for non-primitive Reed-Muller codes, i.e. u-step orthogonalization procedure [18] and Rudolph's one step decoding algorithm in section 3.2 utilize the fact that the null space of their codes contain all u-spaces of $PG(m_1, q)$. Their algorithms can be applied to this code. Some possible improvement of Rudolph's decoding algorithm with be presented in the next section. The guaranteed decodable distance by u-step algorithm is

$$d = 2 + q + \ldots\ldots + q^{m_1 - u} \tag{3.31}$$

which is identical to the BCH bound of the code C in theorem 3.1. The guaranteed decodable distance by one-step algorithm is

$$d_1 = \left\lfloor (q^{m_1} - 1) / (q^u - 1) \right\rfloor + 1 \tag{3.32}$$

d is greater than d_1 in general, but the decoder for one-step decoding algorithm may be simpler than the decoder for u-step decoding algorithm.

In Table 3.1, we list some binary codes from theorem 3.1. n, k are the code length, the number of information digits, d and d_1 are defined in equations (3.31) and (3.32) respectively. The number of information digits of C is identical to the number of information digits of Rudolph's projective geometry code listed in reference [16]. Thus theorem 3.1 establishes the generator polynomials for all Rudolph's projective geometry codes listed in Table 3.1. We will see later that some of the codes in Table 3.1 have more information digits than those of Weldon's non-primitive Reed- Muller codes.

Table 3.1 Binary Cyclic Codes (n, k, d, d_1) Associated with $PG(m_1, 2^s)$

$PG(m_1, 2^s)$	$u = m_1 - 1$	$u = m_1 - 2$	$u = m_1 - 3$	$u = m_1 - 4$
PG(2, 2)	(7, 3, 4, 4)			
PG(3, 2)	(15, 10, 4, 3)	(15, 4, 8, 8)		
PG(4, 2)	(31, 25, 4, 3)	(31, 15, 8, 6)	(31, 5, 16, 16)	
PG(5, 2)	(63, 56, 4, 3)	(63, 41, 8, 5)	(63, 21, 16, 11)	(63, 6, 32, 32)
PG(2, 4)	(21, 11, 6, 6)			
PG(3, 4)	(85, 68, 6, 5)	(85, 24, 22, 22)		
PG(4, 4)	(341, 315, 6, 5)	(341, 195, 22, 18)	(341, 45, 86, 86)	
PG(5, 4)	(1365,1328,6,5)	(1365,1063,22,17)	(1365,483,86,69)	(1365,78,342,342)
PG(2, 8)	(73, 45, 10, 10)			
PG(3, 8)	(585, 520, 10, 9)	(585, 184, 74, 74)		
PG(4, 8)	(4681,4555,10,9)	(4681,3105,74,66)	(4681,590,586,586)	
PG(2, 16)	(273, 191, 18, 18)			
PG(3, 16)	(4369,4112,18,17)	(4369,1568,274,274)		
PG(2, 32)	(1057, 813, 34, 34)			
PG(2, 64)	(4161,3431,66,66)			

We next show that this code contains Weldon's non-primitive Reed-Muller code of corresponding parameters as a subcode.

Let h be an integer less than $q^m - 1$. Let q-ary representation of h be

$$h = \sum_{i=0}^{m_1} h_i q^i \quad ; \quad 0 \leq h_i \leq q-1 \tag{3.33}$$

Let $q = p^s$. Let p-ary representation of h_i be

$$h_i = \sum_{j=0}^{s-1} h_{ij} p^j \; ; \; 0 \leq i \leq m_1 , \quad 0 \leq h_{ij} \leq p-1 \tag{3.34}$$

Let

$$W_j = \sum_{i=0}^{m_1} h_{ij} \text{ ,} \qquad 0 \leq j \leq s\text{-}1 \qquad (3.35)$$

The weight of h over base p is defined as the "digit" sum of the p-presentation of h, that is

$$W_p(h) = \sum_{i=0}^{m_1} \sum_{j=0}^{s-1} h_{ij} = \sum_{j=0}^{s-1} W_j \qquad (3.36)$$

The condition for h such that α^h is a root of the generator polynomial of the projective geometry code specified in theorem 3. 1 is equivalent to the following condition

$$\begin{aligned} w_q(h) &= W_o + p\,W_1 + \ldots\ldots + p^{s-1}\,W_{s-1} = (p^s-1)\,k_o \\ w_q(hp) &= pW_o + p^2W_1 + \ldots\ldots + W_{s-1} \quad = (p^s-1)\,k_1 \qquad (3.37) \\ &\bullet \quad \bullet \quad \bullet \quad \bullet \quad \bullet \quad \bullet \\ w_q(hp^{s-1}) &= p^{s-1}W_o + W_1 + \ldots\ldots + p^{s-2}W_{s-1} = (p^s-1)\,k_{s-1} \end{aligned}$$

where $\qquad 0 \leq k_i \leq (m_1 - u)$

From equation (3. 37), we have

$$(1 + p + \ldots. + p^{s-1})\,(\sum_{j=0}^{s-1} W_j) = (p^s-1)\,(\sum_{j=0}^{s-1} k_j)$$

then

$$\sum_{j=0}^{s-1} W_j = (p-1) \sum_{j=0}^{s-1} k_j \leq s\,(m_1\text{-}u)\,(p\text{-}1) \qquad (3.38)$$

Only binary case is treated explicitly in Weldon's paper [18] . Let g(x) be the generator polynomial of the non-primitive Reed-Muller code of the same parameters $q = 2^s$, m_1 and u as in theorem 3. 1. α^h is a root of

$g(x)$ if and only if h is an integer less than $2^{ms}-1$ such that h is divisible by 2^s-1 and the weight of h to the base 2 is no more than $s(m_1-u)$.

From equation (3.38), we have shown that the non-primitive Reed-Muller code is a subcode of the corresponding projective geometry code defined in theorem 3.1.

In next, we show that the special case $u = m_1 - 1$, Weldon's code is identical to the code defined in theorem 3.1. For p being any prime and $u = m_1 - 1$, Weldon's non-primitive Reed-Muller code contains roots α^h for h is divisible by $q-1$ and

$$w_p(h) = \sum_{j=0}^{s-1} W_j \leq s(p-1) \qquad (3.39)$$

For nonzero h which is a multiple of p^s-1, it is known [18] that $w_p(h)$ is no less than $s(p-1)$. From equation (3.39),

$$w_p(h) = \sum_{j=0}^{s-1} W_j = s(p-1) \qquad (3.40)$$

Since

$$h = \sum_{i=0}^{m_1} h_i q^i = \sum_{i=0}^{m_1} h_i + \sum_{i=0}^{m_1} h_i(q^i-1) \qquad (3.41)$$

that h is a nonzero integer divisible by $q-1$ implies that $\sum_{i=0}^{m_1} h_i$ is also a nonzero integer divisible by $q-1$. Thus

$$w_p\left(\sum_{i=0}^{m_1} h_i\right) \geq s(p-1) \qquad (3.42)$$

But

$$w_p(\sum_{i=0}^{m_1} h_i) \leq \sum_{i=0}^{m_1} w_p(h_i) = w_p(h) = s(p-1) \qquad (3.43)$$

Equations (3.42) and (3.43) imply that

$$w_p(\sum_{i=0}^{m_1} h_i) = \sum_{i=0}^{m_1} w_p(h_i) \qquad (3.44)$$

This implies that

$$W_j = \sum_{i=0}^{m_1} h_{ij} \leq p-1 \ , \quad 0 \leq j \leq s-1 \qquad (3.45)$$

From equations (3.40) and (3.45)

$$W_j = p-1 \ ; \qquad 0 \leq j \leq s-1 \qquad (3.46)$$

For u equals to $m_1 - 1$, Weldon's non-primitive Reed-Muller code contains root α^h for nonzero h satisfying condition (3.46) and for h being equal to zero. These h satisfy the condition (3.37). Weldon's code includes the code defined in theorem 3.1 as a subcode for $u = m_1 - 1$ case. But the former code is a subcode of the latter code in general. Two codes are identical for $u = m_1 - 1$ case.

The number of ways to obtain p - 1 as ordered sum of m nonnegative integers is $\binom{p-1+m-1}{m-1}$, hence the total number of h satisfying equation (3.46) is $\binom{p+m-2}{m-1}^s$. The number of check digits for the new code is

$$r_{m_1-1} = 1 + \binom{p+m-2}{m-1}^s = 1 + \binom{p+m_1-1}{m_1}^s \qquad (3.47)$$

This is an upper bound for the number of check digits for Rudolph's projective geometry code for u being equal to $m_1 - 1$.

For special case $m_1 = 2$, the number of check digits is

$$r_{m_1 -1} = 1 + \binom{p+1}{2}^s \qquad (3.48)$$

In reference[4], Graham and MacWilliams have shown that the number of check digits for any difference-set cyclic code which is identical to Rudolph's projective geometry code for m_1 equal to two and u equal to one is equal to $r_{m_1 -1}$ in equation (3.48). In this case, the non-primitive Reed-Muller code and the code specified in theorem 3.1 are identical to Rudolph's projective geometry code.

The non-primitive Reed-Muller code does not equal to the code defined in theorem 3.1 in general. For p equal to two, the non-primitive Reed-Muller code contains roots α^h for integers h less than $2^{ms}-1$ and satisfying

$$\sum_{j=0}^{s-1} W_j 2^j = \nu (2^s - 1) \quad ; \ \nu \text{ is an integer}$$

$$\sum_{j=0}^{s-1} W_j \le s(m_1 - u) \qquad (3.49)$$

The condition for α^h to be the roots of projective geometry code specified in theorem 3.1 is

$$\sum_{t=0}^{s-1} W_{t+j} 2^t = k_j(2^s - 1) \ ; \ 0 \le k_j \le (m_1 - u) \ , \ 0 \le j \le s-1 \qquad (3.50)$$

where W_{t+j} is equal to W_{t+j-zs} for some z such that $t+j-zs$ is a non-negative integer less than s.

For $m_1 \ge 5$, $q = 2^2$ and $m_1 - u = 3$; $W_o = 0$ and $W_1 = 6$ is a solution to equation (3.49) but not a solution to equation (3.50). In these cases, the non-primitive Reed-Muller codes are proper subcodes of projective geometry codes. It is easy to verify that if a non-primitive Reed-Muller code is a proper subcode of the projective geometry code for $m_1 = m'$, $q = 2^s$, $m_1 - u = \ell$ then the former is also a proper subcode of the latter for $m_1 = m' + i$, $q = 2^s$, $m_1 - u = \ell + i$ for any positive integer i.

In Table 3.2, we list the paramet rs s, ℓ, and m_1 of which the non-primitive Reed-Muller codes are proper subcodes of projective geometry codes specified in theorem 3.1. In the remark column, we give the reason for the former codes being proper subcodes, that is, the W_i's which satisfy the condition (3.49) but not (3.50). Some numerical examples are listed in table 3.3.

Table 3.2 Cases of Which Binary Non-primitive Reed-Muller Codes are Proper Subcodes of Projective Geometry Codes

s	$\ell = m_1 - u$	m_1	Remark (i is any positive integer)
2	$3 + i$	$\geq 5 + i$	$W_o = i$, $W_1 = 6 + i$
3	$2 + i$	$\geq 4 + i$	$W_o = 1 + i$, $W_1 = i$, $W_2 = 5 + i$
3	$3 + i$	$\geq 4 + i$	$W_o = i$, $W_1 = 4 + i$, $W_2 = 5 + i$
4	$2 + i$	$\geq 5 + i$	$W_o = 2 + i$, $W_1 = 1 + i$, $W_2 = 4 + i$, $W_3 = 5 + i$

Table 3.3 Numerical examples of codes in Table 3.2

s	ℓ	m_1	(n, k_1)	(n, k_2)
2	3	5	(1365, 481)	(1365, 483)*
2	3	6	(5461, 3143)	(5461, 3185)
2	4	6	(5461, 742)	(5461, 1036)
2	3	7	(21845, 17532)	(21845, 17588)
2	4	7	(21845, 9048)	(21845, 9096)
3	2	4	(4681, 3090)	(4681, 3105)*
3	3	4	(4681, 575)	(4681, 590)*
4	2	3	(4369, 1505)	(4369, 1568)*
4	2	4	(69905, 50779)	(69905, 52079)
4	3	4	(69905, 4979)	(69905, 5579)

k_1 is the number of information digits of non-primitive Reed-Muller code. k_2 is that of projective geometry code.

A code containing the code of theorem 3.1 as a subcode and also satisfying the parity checks associated with u-dimensional spaces will be presented. We prove that in some special cases, this code is identical to Rudolph's projective geometry code.

From equations (3.37) and equation (3.46), the code C_{m_1-1} specified in theorem 3.1 for u equal to $m_1 - 1$ has its generator polynomial containing α^h as roots if and only if

* indicates this projective geometry code appears in Table 3.1.

$$h = 0 \tag{3.51}$$

or h satisfying the condition

$$w_q\,(h\,p^j\,) = q-1 \;\;;\quad 0 \leq j \leq s-1 \tag{3.52}$$

Let C_u be a code whose generator polynomial consists of roots α^h which is a product of the roots $\alpha^{h^{(i)}}$ $(1 \leq i \leq m_1 - u)$ of the generator polynomial of C_{m_1-1}. $\alpha^{h^{(i)}}$ are not necessarily distinct.

$$h = \sum_{i=1}^{m_1-u} h^{(i)} \tag{3.53}$$

From equations (3.51), (3.52) and (3.53)

$$w_q(h p^j) = w_q\,(\sum_{i=1}^{m_1-u} h^{(i)}\,p^j\,) \leq \sum_{i=1}^{m_1-u} w_q(h^{(i)} p^j) \leq (m_1-u)\,(q-1) \tag{3.54}$$

$h^{(i)}$ are multiples of q-1, then h is a multiple of q-1. From equation (3.54),

$$w_q(h p^j) = \nu_j\,(q-1) \;\;;\quad 0 \leq j \leq s-1\,,\quad 0 \leq \nu_j \leq (m_1-u)\,(q-1) \tag{3.55}$$

Hence the code C_u contains the code specified in theorem 3.1 as a subcode. We want to show that C_u has all u-dimensional flats as parity checks. Furth.rmore C_u is identical to a Rudolph's projective geometry code when C_{m_1-1} is equal to a Rudolph's projective geometry code.

The generator matrix of the dual code of C_{m_1-1} is

$$G_{m_1-1} = \left[1 \;\; \alpha^h \;\; \text{------} \;\; (\alpha^h)^{n-1}\right] \tag{3.56}$$

for h equal to zero or h satisfying

$$W_j = p - 1 \qquad ; \qquad 0 \leq j \leq s-1 \tag{3.57}$$

where W_j is defined in equation (3.35).

Let $(\ell_0, \ell_1, \ldots, \ell_{m_1})$ be corresponding to h if

$$\ell_i = h_i \quad , \qquad 0 \leq i \leq m_1 \tag{3.58}$$

Then the row space of G_{m_1-1} can be shown to be identical to the row space of

$$L_{m_1-1} = \left[u_0^{\ell_0} \; u_1^{\ell_1} \; \text{--------} \; u_{m_1}^{\ell_{m_1}} \right] \tag{3.59}$$

with $(\ell_0, \ell_1, \ldots, \ell_{m_1})$ corresponding to the h equal to zero or h satisfying equation (3.57) as follows.

$$(\alpha^h)^j = (\alpha^j)^h = \left(\sum_{i=0}^{m_1} v_{ij} \alpha^i \right)^{\sum_{t=}^{m_1} \sum_{k=0}^{s-1} h_{tk} p^{ts+k}} \tag{3.60}$$

$$(\alpha^h)^j = \prod_{t=0}^{m_1} \prod_{k=0}^{s-1} \left(\sum_{i=0}^{m_1} v_{ij}^{p^k} \alpha^{ip^{ts+k}} \right)^{h_{tk}} \tag{3.61}$$

The rest of the proof can be accomplished by analogy of the proof of lemma 3.2.

Let L'_u be a matrix whose row vectors are the vector product of the row vectors of L_{m_1-1} taking m_1 - u at a time.

The parity check matrix of C_u can be written of the form in equation (3.56) for h satisfying equation (3.53). Similar argument as u equal to m_1 -1 case, the matrix L'_u is also a parity check matrix of C_u.

Since any $(m_1 - 1)$-space of $PG(m_1,q)$ is in the row space of L_{m_1-1}. Any u-space can be considered as an intersection of some $m_1 - u$ number of $(m_1 - 1)$-spaces, hence it must be in the row space of L'_u. In case that the vectors associated with $(m_1 - 1)$-space of $PG(m_1,q)$ span the row space of L'_u, the vectors associated with u-spaces of $PG(m_1,q)$ also span the row space of L'_u.

Thus the code C_u has all u-spaces of $PG(m_1,q)$ as its parity checks. C_u is equal to Rudolph's projective geometry code provided C_{m_1-1} is equal to Rudolph's projective geometry code.

3.4 On Rudolph's Decoding Algorithm for Projective Geometry Codes

Rudolph's Decoding Algorithm uses all u-spaces of $PG(m_1,q)$ for majority voting. In this section, we show by an example that it is possible to choose a set of u-spaces which are orthogonal on a point for majority voting and achieve the same guaranteed decodable distance by Rudolph's method in some cases.

Example. For $m_1 = 4$, $q = 2$, $u = 2$, we have binary (31, 15) code. The Rudolph decoding algorithm requires all 105 2-spaces to be used for majority voting and achieve a guaranteed decodable distance 6.

Let α be a root of primitive polynomial $x^5 + x^2 + 1$ over $GF(2)$, then the matrix G in equation (3.6) becomes

$$\begin{bmatrix} 0000100101100111110001101110101 \\ 0001001011001111100011011101010 \\ 0010010110011111000110111010100 \\ 0100001001011001111100011011101 \\ 1000010010110011111000110111010 \end{bmatrix} \tag{3.62}$$

We can choose the following five 2-spaces orthogonal to the first point.

$$\begin{bmatrix}111001000000100000001100000000000000\\100110000001000000000000001000100010\\100000011000000000010000000101001000\\10000000011000000010001000000010100\\100000000000000111001000000010000001\end{bmatrix} \qquad (3.63)$$

The guaranteed decodable distance by using this set of orthogonal parity checks is also 6.

It is impossible to choose sufficient number of u-spaces which are orthogonal on a point for majority voting and achieve the same guaranteed distance by Rudolph's method in general. For example, any two $(m_1 - 1)$-spaces of $PG(m_1,q)$ intersect a $(m_1 - 2)$-space. We cannot get a set of two or more parity checks orthogonal on a point for $m_1 > 2$. The guaranteed decodable distance by Rudolph's method is

$$\left[\frac{q^{m_1}-1}{q^{m_1-1}-1}\right]+1 = \left[q + \frac{q-1}{q^{m_1-1}-1}\right]+1 = q+1$$

Thus for projective geometry code associated with $(m_1 - 1)$ - spaces of $PG(m_1,q)$ with $m_1 > 2$, we cannot obtain $q(q \geq 2)$ number of $(m_1 - 1)$ spaces orthogonal on a point for majority voting, hence we cannot achieve the same guaranteed decodable distance $q + 1$.

IV. INVESTIGATION OF THRESHOLD DECODING FOR CYCLIC CODES

Since BCH codes are most powerful random error-correcting codes, we investigate whether all BCH codes can be L-step orthogonalized. Unfortunately, we find that a class of double error-correcting BCH codes cannot be L-step orthogonalized. On the other hand, we found that BCH codes with length q^m-1 as well as Euclidean geometry codes can be one step decoded by parity checks which are not necessary orthogonal. We cannot decode these codes to their minimum distances in general. These codes decoded by this method is comparable to projective geometry codes decoded by Rudolph's method. A comparison is made for the codes derived from projective geometries and the codes from Euclidean geometries by u-step decoding method. For the same error-correcting ability, the transmission rate increases as code length increases but the decoder complexity also increases.

4.1 Non-Orthogonality of Some BCH Codes

Massey [13] in his earlier work suggested an important area of research to be investigation of L-step orthogonalization procedure for block linear codes. An interesting result is obtained in this direction. That is, some double error-correcting BCH codes cannot be L-step orthogonalized. The proof essentially consists of showing no set of d-1 (where d is the minimum distance of the code) parity checks orthogonal on any noise bit or sum of noise bits, can be formed. We first represent a necessary condition for a code to be L-step orthogonalized as follows.

Lemma 4.1 Let $g_o(x)$ be the generator polynomial of a binary code C_o and $g_o(1) \neq 0$. Let C_{oe} be an extension code of C_o obtained by adding an overall parity check as its first digit to C_o. If C_{oe} is invariant under a transitive permutation group, a necessary condition for C_o to be L-step orthogonalized is that

$$\frac{n+1}{2} \geq \left(\frac{3}{2} - \frac{1}{d-1} \right) d' \qquad (4.1)$$

where n is the code length, d is the minimum distance of the code C_o. d' is the minimum distance of the dual code C of C_o.

Proof: Let $x_1, x_2, \ldots\ldots\ldots x_{d_o-1}$ (d_o is an odd integer no more than d) be the set of vectors in the code C which are used to form a set of d_o-1 parity checks orthogonal on a selected sum of noise bits $e_{i_1} + e_{i_2} + \ldots + e_{i_y}$ where e_{i_y} are distinct noise bits $(1 \leq i_y \leq n)$. Let x_o be a vector which is a sum of $x_1, x_2, \ldots, x_{d_o-1}$ and the vector with all one entries. Let w_i $(0 \leq i \leq d_o-1)$ be the weight or the number of 1's of the vectors x_i respectively. Since d_o-1 is even, x_o must have ones in the positions $i_1, i_2, --, i_y$. It is easy to verify that

$$\sum_{i=0}^{d_o-1} (w_i - y) = n - y \qquad (4.2)$$

Without loss of generality, let

$$w_1 \leq w_2 \leq ---- \leq w_{d_o-1} \qquad (4.3)$$

From equation (4.2) and (4.3), we have

$$y \geq \frac{w_o + (d_o-1)\, w_1 - n}{d_o-1} \qquad (4.4)$$

We now want to show that

$$\frac{w_o + w_1 - (d' - 1)}{2} \geq y \qquad (4.5)$$

and

$$w_o \leq n\text{-}d' \tag{4.6}$$

in order to prove this lemma. Let C'_o be a code generated by $(x-1)\, g_o(x)$ where $g_o(x)$ is the generator polynomial of C_o. Let C' be the dual of C_o. C' is a code contains the code C as a subcode. C' contains the vector with all one entries, hence C' contains x_o and $x_o + x_1$ as code words. The weight of $x_o + x_1$ is $w_o + w_1 - 2y$. Equations (4.5) can be established if we show that the minimum distance of the code C' is d'-1. C is a subcode of C' and C contains all the code words of C' which have even weight. It is easy to verify that the extension code C'_e of C' obtained by adding an overall parity check as its first digit to the code C' is a dual code of C_{oe}. C_{oe} is a code which is invariant under a transitive permutation group implies that its dual code also invariant under the same transitive permutation group (c.f. Theorem 11.1 of reference [14]). d' is the minimum distance of C, then d' is an even integer and d' is also the minimum distance of C'_e. Since C'_e is invarient under a transitive group, there exists a vector v_e in C'_e such that the first digit of v_e is nonzero and the weight of v_e is d'. The vector v obtained by deleting the first digit of v_e has weight equal to d'-1 and v is a code word of C' with minimum weight d'-1. Hence equation (4.5) can be established. The weight of x_o must be odd. Since the minimum even distance of C' is d', the largest possible weight of x_o is n-d'. Hence equation (4.6) is established.

From equations (4.4) and (4.5)

$$\frac{w_o + w_1 - (d'-1)}{2} \geq y \geq \frac{w_o + (d_o - 1)\, w_1 - n}{d_o - 1} \tag{4.7}$$

Rearrange equation (4.7), we have

$$\left(\frac{1}{2} - \frac{1}{d_o - 1} \right) w_o \geq \frac{w_1}{2} - \frac{n}{d_o - 1} + \frac{d' - 1}{2} \qquad (4.8)$$

$$w_1 \geq d' \qquad (4.9)$$

Substituting equations (4.6) and (4.9) into equation (4.8), we have

$$\left(\frac{1}{2} - \frac{1}{d_o - 1} \right) (n - d') \geq \frac{d'}{2} - \frac{n}{d_o - 1} + \frac{d' - 1}{2} \qquad (4.10)$$

Rearranging equation (4.10), we have

$$\frac{n+1}{2} \geq \left(\frac{3}{2} - \frac{1}{d_o - 1} \right) d' \qquad (4.11)$$

From the property that C_e' is invariant under a transitive permutation group, we have shown the minimum distance of C' is odd (d'-1 is odd). Similarly, we can show d is an odd integer since C_{oe} is invariant under a transitive permutation group. d_o must equal to d when C_o can be L-step orthogonalized. The lemma is proved by equation (4.11).

From equation (4.11)

$$J = d_o - 1 \leq \frac{2d'}{3d' - n - 1} \qquad (4.12)$$

The maximum number of parity checks orthogonal on any noise bit or sum of noise bits is no more than J.

Let α be a primitive root of $GF(2^m)$. A binary NBCH code C_o is defined to be the code consists of $\alpha^1, \alpha^2, \ldots, \alpha^{d-1}$ consecutive roots. The extension code C_{oe} obtained by adding an overall parity check to C_o as its first digit is invariant under a transitive group. Lemma 4.1 is applicable to this code.

Theorem 4.1 All double error-correcting binary NBCH codes C_0 cannot be L-step orthogonalized for $m \geq 7$.

Proof: The minimum distance of these code is at least 5. Let $d_0 = 5$, and using equation (4.11), we have

$$\frac{(2^m - 1) + 1}{2} \geq \left(\frac{3}{2} - \frac{1}{5-1} \right) d' \tag{4.13}$$

or $\quad 2^{m+1} \geq 5 d'$

For m to be odd, d' is equal to $2^{m-1} - 2^{(m+1)/2-1}$ [8]. Equation (4.13) becomes

$$2^{m+1} \geq 5(2^{m-1} - 2^{(m+1)/2-1})$$

or $\quad 5 \cdot 2^{(m+1)/2-1} \geq 2^{m-1}$

This condition cannot be satisfied for m greater than or equal to 7.

For m to be even, d' is equal to $2^{m-1} - 2^{(m+2)/2-1}$. Equation (4.13) becomes $2^{m+1} \geq 5\,(2^{m-1} - 2^{(m+2)/2-1})$ or $5 \cdot 2^{m/2} \geq 2^{m-1}$.

This condition cannot be satisfied for m greater than or equal to 8.

Thus we have proved the theorem.

In this theorem, we have shown that all double error-correcting NBCH codes C_0 cannot be L-step orthogonalized for $m \geq 7$. In next, we show that the binary double error-correcting NBCH code C_0 cannot be L-step orthogonalized for m equal to 5. The dual C of C_0 has vectors of the following four possible weights 0, 12, 16, 20 [8]. We use same notation as in the previous lemma and theorem. A vector in the code C' but not in C has weight equal to 11, 15, 19, 31.

$$(w_o - y) + (w_1 - y) + (w_2 - y) + (w_3 - y) + (w_4 - y) = n - y \qquad (4.14)$$

$$(w_o - y) + (w_1 - y) \geq 11 \qquad (4.15)$$

$$(w_1 - y) + (w_2 - y) \geq 12 \qquad (4.16)$$

Let $w_1 \leq w_2 \leq w_3 \leq w_4$, equation (4.14) implies

$$y \geq (w_o + w_1 + 3w_2 - 31)/4 \qquad (4.17)$$

Equations (4.15) and (4.16) imply that

$$y \leq (w_o + w_1 - 11)/2 \qquad (4.18)$$

$$y \leq (w_1 + w_2 - 12)/2 \qquad (4.19)$$

With the restrictions that $w_1 \leq w_2$, w_i $(i=1,2)$ must equal 12, 16, or 20, w_o must equal to 11, 15, or 19, one can verify that there does not exist a positive integer y such that equations (4.17), (4.18) and (4.19) are satisfied simultaneously. This shows that C_o cannot be L-step orthogonalized for $m=5$.

4.2 One-Step Majority Decoding of Some Cyclic Codes

In this section, we show a method to decode all cyclic codes whose extension codes are invariant under a doubly transitive permutation group and also to decode the cyclic codes which have an addition parity check bit than the previous ones.

Let C_o be a q-ary code with length $n=q^m-1$ and let $q=p^s$ where p is a prime number. The extended code C_{oe} of C_o is a code with an overall parity check to C_o as its first digit. The first position of a code vector in C_{oe} is numbered 0, the i-th position for $i > 1$ is numbered α^{i-2} where α is a primitive element in $GF(q^m)$. Thus the q^m positions of a code vector is numbered by q^m elements in $GF(q^m)$. An affine transformation with parameters a, b belong $GF(q^m)$, $a \neq 0$ is a permutation which carries the symbols in position X to position $aX + b$. Such a transformation can be applied to any extended code associated with a primitive element of $GF(q^m)$. A code will be called invariant under the affine group if every affine permutation carries every code word into another code word. The necessary and sufficient condition for a code to be invariant under the affine group is as follows [6].

Let g(x) be the generator polynomial of C_o. Let i be a positive integer less than q^m. Let J(i) be the set of nonzero integers j such that each coefficient of the p-ary representation of i is greater than or equal to the corresponding one of j. The extended code C_{oe} is invariant under the affine group of permutations if and only if for every α^i which is a root of the generator polynomial g(x), for every j belongs J(i), α^j is also a root of g(x) and $g(1) \neq 0$. Let C be the dual of C_o. Let h(x) be the generator polynomial of C. Let $h^*(x)$ be the reciprocal polynomial of h(x). Then

$$h^*(x)\, g(x) = x^{q^m - 1} - 1$$

(x-1) is not a factor of g(x) implies that x-1 is a factor of h(x). Let C' be a code generated by h(x) / (x-1), then the extension code C'_e is invariant under the affine group of transformations.

We are concerned with the decoding of the codes C and C'. Kasami et al [7] have shown the connection between binary codes which are invariant under a doubly transitive group and their connection with balanced incomplete block design. Rudolph has decoded projective geometry codes by using the property of balanced incomplete block design. The argument to prove the following Lemma is similar to their argument.

Lemma 4.2 Let n be the code length of the code C. Let k_1 be the minimum weight of a vector in C_{oe} where C_{oe} is an extension code of C_0 which is a dual of C. C can be decoded by one step decoding to the distance of $[n/(k_1 - 1)] + 1$.

Proof: Let v be a vector of minimum weight k_1 in C_{oe}. Let E be the equivalence class of v under the doubly transitive permutation group. Let A be a matrix whose rows are vectors in E. Since the permutation group is doubly transitive, there exists a permutation which will permute i-th column of A to the i_1-th column of A and the j-th column of A to the j_1-th column of A. Since the permutation leaves the rows of A invariant except it rearranges the rows of A, it follows that the numbers of nonzero entries in the i_1-th column and i-th column are equal. The number of nonzero entries in corresponding positions i and j columns are equal to the nonzero entries in i_1 and j_1 columns. Let the number of nonzero elements in any column be r and the number of nonzero elements in corresponding positions of any two columns be λ. Let A_1 be the matrix obtained by deleting the first column of A. Let A_2 be the matrix obtained by deleting all rows of A_1 whose leftmost entries are zero. A_1 is a r by n matrix with all entries in the first column nonzero and exactly λ number of nonzero entries in any other column. The row vectors of A_1 are parity checks of the code C. Hence C can be decoded to the distance $[r/\lambda] + 1$. If we change all the nonzero entries of the matrix A into 1, the new matrix is an incidence matrix of a balance incomplete block design.

Thus

$$n/(k_1 - 1) = r/\lambda.$$

We have proved the lemma.

In next, we show how to decode a code C' which has one more information digit than that of C.

Lemma 4.3 Let C' be a cyclic code whose generator polynomial is h(x) / (x-1) where h(x) is the generator polynomial of the code C in lemma 4.2. C' can be decoded to the distance $\left[n/(k_1 - 1)\right]$ by one step decoding method.

Proof: Let the dual of C' be C'_0, then C'_0 must be subcode of C_0. We want to show that the row vector of the matrix A_1 (defined in the proof of lemma 4.2) is in the code C'_0 if and only if the overall parity check bit added to this row vector is equal to zero. The total number of such vectors is $r - \lambda$ because the number of nonzero elements in corresponding positions of the first two columns of the matrix A is λ. Consider the $(r - \lambda)$ x n submatrix of A_1 whose rows consist of these $r - \lambda$ vectors, all entries in the first column is nonzero and at most λ number of nonzero entries in any other column. We can decode C' to the distance $\left[n/(k_1 - 1)\right]$ by the same argument of lemma 4.2 provided all row vectors of the submatrix are in C'_0. We prove this as follows. The generator matrix of C'_0 had the form

$$M_1 = \begin{bmatrix} 1 & \alpha^{e_1} & \cdot & \cdot & \cdot & (\alpha^{e_1})^n \\ & \cdot & \cdot & \cdot & \cdot & \\ 1 & \alpha^{e_i} & \cdot & \cdot & \cdot & (\alpha^{e_i})^n \end{bmatrix} \qquad (4\ 20)$$

where e_i are positive integers.

The generator matrix of C_{oe} must equal to

$$M_2 = \begin{bmatrix} 1 & 1 & 1 & \cdot\cdot & 1 \\ 0 & 1 & \alpha^{e_1} & \cdot\cdot & (\alpha^{e_1})^n \\ & \cdot & \cdot & \cdot & \cdot \\ 0 & 1 & \alpha^{e_i} & \cdot\cdot & (\alpha^{e_i})^n \end{bmatrix} \qquad (4.21)$$

because the generator matrix of C_0 is the matrix obtained by deleting the first column of the matrix M_2 and the element in the first column is indeed an overall parity check. Any vector v in C_{oe} whose first digit is equal to zero can be obtained by linear combination of the vectors from the second row to the last row of the matrix M_2 and vice versa. By deleting the first row and the first column of the matrix M_2, we can obtain the matrix M_1. The row vector of A_1 is in the code C'_0 if and only if the overall parity check bit added to this row vector is equal to zero. This proves the lemma.

The obvious reason to choose the vector v of the minimum weight k_1 in C_{oe} is that we want the integer $\left[n/(k_1 - 1)\right]$ to be as large as possible for a fixed n.

The extension codes of NBCH codes are known to be invariant under the doubly transitive affine group. The minimum distances of some binary NBCH codes are known from references [15], we can apply lemmas 4.2 and 4.3 to these codes. Let n, k, d_1 denote the code length, the number of information digits and the guaranteed decodable distance by one step decoding method. Some binary BCH code with parameters (n, k, d_1) are listed in table 4. 1.

Recall that the Euclidean geometry code C with parameters m, u, q is defined as the cyclic code whose generator polynomial contains α^h as roots for h satisfying the condition $0 \leq w_q(hp^j) \leq (m-u)(q-1)$ for $0 \leq j \leq s-1$ where α is a primitive element of $GF(q^m)$ and $q = p^s$. The dual C_0 of the Euclidean geometry code C has the generator polynomial contains α^{hp^j} as roots for h satisfying the condition $0 < w_q(h) < u(q-1)$. The

Table 4.1 Binary BCH Codes by One Step Decoding Method

n,	k,	d_1	n,	k,	d_1
31	10	7	31	11	6
63	9	22	63	10	21
63	15	13	63	16	12
63	23	10	63	24	9
63	45	5	63	46	4
127	14	26	127	15	25
255	12	86	255	13	85
255	16	52	255	17	51
255	20	52	255	21	51
511	18	103	511	19	102
1023	15	342	1023	16	341
1023	20	205	1023	21	204

extension code C_{oe} of the code C_o satisfies the necessary and sufficient condition for a code to be invarient under the doubly transitive affine group. The C_{oe} contains all u-dimensional flats of EG(m, q). The weight of a vector v associated with an u-dimensional flat is q^u. The q-ary (m-u)(q-1)-th order Reed-Muller code contains q-ary code C_{oe} as a subcode. The minimum distance of the q-ary (m-u) (q-1)-th order Reed-Muller code is q^u[6]. Thus the minimum weight k_1 of the vector v in C_{oe} is q^u. We can apply lemma 4.2 to Euclidean geometry code C and the guaranteed decodable distance is

$$d_1 = \left[(q^m - 1) / (q^u - 1)\right] + 1 \qquad (4.22)$$

The modified Euclidean geometry code has one more information digit than the corresponding Euclidean geometry code. The guaranteed decodable distance is $\left[(q^m - 1) / (q^u - 1)\right]$ by lemma 4.3.

4.3 Comparisons and Remarks

The guaranteed decodable distance of Euclidean geometry code C by one-step decoding is $d_1 = \left[(q^m - 1) / (q^u - 1)\right] + 1$ which is less than or equal to the guaranteed decodable distance d by u-step decoding.

$$d = 2 + q + \dots + q^{m-u} \tag{4.23}$$

because

$$2 + q + \dots + q^{m-u} = 1 + \frac{q^m - q^{u-1}}{q^u - q^{u-1}} \geq 1 + \left[\frac{q^m - 1}{q^u - 1}\right] = d_1$$

When u is not equal to one, d is greater than d_1. The decoder for one-step decoding method may be simpler than the decoder for u-step decoding method. For example, when u equal to m-1 the guaranteed decodable distance of Euclidean geometry code C is 1 + q for one-step decoding method but 2 + q for (m-1)-step orthogonalization procedure. The decoder for the former method is simpler than the decoder for the latter method. The guaranteed decodable distances for a Euclidean geometry code and a projective geometry code with same parameters are both equal to d for u-step decoding method and both equal to d_1 for one-step decoding method. We list Euclidean geometry codes and projective geometry codes together for comparison purposes as Table 4.2. In this table, q, m, u are the parameters of the Finite geometry codes. n and k under the column PG(or EG) are the code length and the number of information digits of projective geometry code (or Euclidean geometry code) respectively. The guaranteed decodable distances d by u-step decoding method and d_1 by 1 step decoding method are listed in the last two columns of table 4.2.

Table 4.2 Comparison of Projective Geometry Codes and Euclidean Geometry Codes

			PG		EG		u-step	1-step
q	m	u	n	k	n	k	d	d_1
2^2	2	1	21	11	15	6	6	6
	3	2	85	68	63	47	6	5
	3	1	85	24	63	12	22	22
	4	3	341	315	255	230	6	5
	4	2	341	195	255	126	22	18
	4	1	341	45	255	20	86	86
	5	4	1365	1328	1023	987	6	5
	5	3	1365	1063	1023	747	22	17
	5	2	1365	483	1023	287	86	69
	5	1	1365	78	1023	32	342	342
2^3	2	1	73	45	63	36	10	10
	3	2	585	520	511	447	10	9
	3	1	585	184	511	138	74	74
	4	3	4681	4555	4095	3970	10	9
	4	2	4681	3105	4095	2584	74	66
	4	1	4681	590	4095	405	586	586
2^4	2	1	273	191	255	174	18	18
	3	2	4369	4112	4095	3839	18	17
	3	1	4369	1568	4095	1376	274	274
2^5	2	1	1057	813	1023	780	34	34
2^6	2	1	4161	3431	4095	3366	66	66

Let the code length, the number of information digits and the number of check digits of the Euclidean geometry code with parameters q, m (m=m'), u be n_e, k_e and r_e respectively. Let those of the projective geometry code with same parameters be n_p, k_p and r_p respectively. Let those of the projective geometry code with same q and u but m=m'-1 be n'_p, k'_p and r'_p respectively. From Table 4.2, we observed that for m'-1 > u

$$k_e = k_p - k'_p - 1 \tag{4.24}$$

Since $n_e = q^{m'} -1$, $n_p = 1 + q + \ldots + q^{m'}$ and $n'_p = 1 + q + \ldots + q^{m'-1}$ therefore

$$n_e = n_p - n'_p - 1 \tag{4.25}$$

From equations (4.24) and (4.25), the parity checks of these codes satisfy the relation

$$r_e = r_p - r'_p \tag{4.26}$$

We observed that the number of parity check r_e when m'-1 equal to u is

$$r_e = \binom{p+m'-1}{m'}^s \tag{4.27}$$

These observations can be explained as follows. The integer r_p is equal to the number of distinct elements in the set A which consists of integers a such that

$$0 \leq a < q^{m'+1}-1$$

$$w_q(a\,p^j) = \nu_j\ (q-1)\,,\quad 0 \leq j \leq s-1,\quad 0 \leq \nu_j \leq (m'-u) \tag{4.28}$$

(c.f. theorem 3. 1)

The integer r'_p is equal to the number of distinct elements in the set B which consists of integers b such that

$$0 \leq b < q^{m'}-1$$

$$w_q(\,b\,p^j) = \nu_j\,(q-1);\ 0 \leq j \leq s-1,\ 0 \leq \nu_j \leq (m'-1-u) \tag{4.29}$$

The integer r_e is equal to the number of distinct elements in the set C which consists of integers c such that

$$0 \leq c < q^{m'-1}$$
$$0 \leq w_q(c\, p^j) \leq (m'-u)(q-1), \quad 0 \leq j \leq s-1 \tag{4.30}$$

(c.f. theorem 2.1)

Clearly the set B is a subset of A. We first show that

$$r_e \leq r_p - r'_p \tag{4.31}$$

by showing for a distinct c in the set C, there corresponds a distinct h in the set A but not the set B. Let

$$c = \sum_{i=0}^{m'-1} h_i q^i \ , \qquad 0 \leq h_i \leq q-1 \tag{4.32}$$

be an integer in C. Let

$$h = c + h_{m'} q^{m'} \tag{4.33}$$

where $h_{m'}$ is as follows.

In case (1), $w_q(c)$ is not a multiple of (q-1). Let $h_{m'}$ be a positive integer less than q-1 and $h_{m'} + w_q(c)$ is a multiple of (q-1). In case (2), $w_q(c)$ is a multiple of (q-1), then

$$w_q (c\, p^j) = \nu_j (q-1) \ , \quad 0 \leq j \leq s-1, \quad 0 \leq \nu_j \leq (m'-u) \tag{4.34}$$

If ν_j is less than (m'-u) for all j, let $h_{m'}$ equal to q-1. If there exist one j such that ν_j is equal to (m'-u), let $h_{m'}$ equal to zero.

It is easy to verify that the integer h defined in equation (4.33) is in the set A but not the set B. Two different elements in the set C will correspond to two different elements in the set A but not the set B. Hence equation(4.31) is established. For any element h' is the set A but not the set B, h' can be written as

$$h' = \sum_{i=0}^{m'} h_i' q^i , \qquad 0 \le h_i' \le q-1 \tag{4.35}$$

It is easy to verify that

$$c' = \sum_{i=0}^{m'-1} h_i' q^i \tag{4.36}$$

is in the set C. Hence

$$r_e \ge r_p - r_p' \tag{4.37}$$

Thus equation (4.26) is established.

When $m'-1$ equals to u,

$$r_p = 1 + \binom{p+m'-1}{m'}^s \tag{4.38}$$

(c.f. equation (3.47)) and

$$r_{p'} = 1$$

Hence equation (4. [illegible]) is established.

We list the transmission rate (k/n) of some codes listed in Table 4.2 with same guaranteed decodable distance d in Table 4.3.

In table 4.3, u is the number of steps required for decoding. The quantity in parentheses is k/n. For same d, the transmission rate increases. The decoder complexity increases as the code length, the number of information digits, the number of steps required for decoding increase.

Table 4. 3 Transmission Rate of Binary Finite Geometry Codes

u d_u	1	1	2	2	3	3
6	0.4 (6/15)	0.524 (11/22)	0.715 (45/63)	0.8 (68/85)	0.9 (230/255)	0.922 (315/341)
10	0.571 (36/63)	0.644 (47/73)	0.874 (447/511)	0.89 (520/585)	0.97 (3970/4095)	0.972 (4555/4681)
18	0.684 (174/255)	0.7 (191/273)	0.938 (3839/4095)	0.942 (4112/4369)		
22	0.192 (12/63)	0.282 (24/85)	0.494 (126/255)	0.572 (195/341)	0.73 (747/1023)	0.78 (1063/1365)
74	0.27 (138/511)	0.314 (184/585)	0.655 (2584/4095)	0.673 (3105/4681)		
86	0.0785 (20/255)	0.132 (45/341)	0.28 (287/1023)	0.354 (483/1365)		

V. APPLICATION OF CODING THEORY TO INFORMATION RETRIEVAL

5.1 Introduction

An entry in theindex file for the document collection typically includes an identification number for the document together with a list of descriptors or attributes characterizing that particular document. The descriptors are commonly chosen from a dictionary and an upper bound is placed on the number of descriptors which may be chosen to characterize any single document. A "query" to such a collection is again a list of descriptors from the dictionary. A typical dictionary might contain a number of N descriptors between 10^3 and 10^4 and the maximum number of descriptors would normally fall between 5 and 10 [9].

The information retrieval problem considered here may be defined as follows: Given a query, we wish to devise a process by obtaining a list of documents such that each of these has all the descriptors possessed by the query. In other to automate the retrieval process, it is necessary to encode both document and query data in some form suitable for automatic process. Two important methods for doing this have been proposed. One is using zero-false-drop codes proposed by Kautz and Singleton [9]. Another one is derived from algebraic coding theory by Chien and Frazer [3]. Encoding by the former method usually has longer digit representation than the latter method. The time required to retrieval is comparatively less by using zero-false-drop code. We now give a brief summary of these two codes. A zero-false-drop code of order t (ZFD_t) is a set of n-digit binary code words satisfying the property that every sum of up to t different code words logically include no other code word where the sum of the n-digit binary words is their digit by digit Boolean sum. Let each of the N descriptors in the dictionary be assigned a unique n-digit binary code word of ZFD_t code. Each document is represented by an n-digit word which is obtained by forming the digit by digit Boolean sum of the code words of all of its constituent descriptors. The query is represented in identical fashion. It follows directly from the property of the ZFD_t code that as long as no more than t descriptors are associated with any one document, the query is logically included in a particular document word if and only if all of the query descriptors are included among the descriptors associated with the document. Thus ZFD_t code can be used for information retrieval file and guarantees no false drop.

The method of encoding documents and queries from the algebraic structure of linear error-correcting code are as follows:

Let V_1 be a linear code with t-error-correcting ability. Let H be the parity check matrix of V_1. The row space of H is in the null space V_2 of V_1. The syndrome of a vector v is defined to be vH^T where H^T is the transpose of the matrix H. If the code length of V_1 is N_1 which is greater than N, the total number of descriptors in the dictionary, we can represent each descriptor as a column of H; then each descriptor is represented by binary n-tuples where n is the number of the parity check in the code. A document (or a query) is represented by mod-two linear combination of the n-tuples each of which corresponds to a constituting descriptor of the document. A query is represented in a similar manner. We limit the maximum number of descriptors to characterize each document to t. The documents can be represented un-ambiguously because no two distinct linear combinations of t or fewer columns of H are equal. The retrieval method is based on the following argument. In an error-correcting code, we usually choose the vector with minimum weight in the coset as a coset leader. A coset leader

C_d is said to cover a cost leader C_q if C_d contains 1 whenever C_q contains 1 in any digit position. C_d covers C_q if and only if $w(C_d + C_q) + w(C_q) = w(C_d)$ where $w(x)$ = weight of the coset leader x. We can consider each digit position of a coset leader corresponding to a descriptor in the dictionary. A coset leader C_d corresponds to a document if C_d has 1's in the digit positions corresponding to constituting descriptors of the document, and has O's in all other digit positions. A coset leader C_q corresponding to a query in similar way. A document C_d covers a query if and only if $w(C_d + C_q) + w(C_q) = w(C_d)$. In information retrieval, we do not have the coset leader C_d and C_q explicitly available to us for testing. Instead we have syndromes s_d and s_q of C_d and C_q respectively. From previous discussion, it is clear that the main computational problem in our retrieval process is that of determining whether $f(s_{d+q}) = f(s_d) - f(s_q)$ where $f(s_x)$ is defined to be equal to $w(x)$. In coding terminology, computing $f(s_{d+q})$ is equivalent to finding the weight of the coset leader from the syndrome of the coset. Details of a retrieval method derived from algebraic BCH code decoding can be found in reference [3].

5.2 Zero-False-Drop Codes Derived From Finite Geometries

Two classes of zero-false-drop codes can be constructed from finite geometries. One is derived from projective geometries and the other one is derived from Euclidean geometries.

Kautz and Singleton [9] have derived a bound for the order of zero-false-drop code of constant weight codes. Let w be the weight of any code word. Let μ be the dot product of any pair of code words. Let μ_{max} be the maximum number of such μ's. If $w \geq t\mu_{max} + 1$, then any code word cannot possibly be contained in the sum of any t other code words since it overlaps each of these other code words in nc more than $\mu_{\max}$ positions. The constant weight code has order t as a zero false drop code such that

$$t \geq \left[\frac{w - 1}{\mu_{\max}} \right] \tag{5.1}$$

where the bracket indicates "the integer part of".

The projective geometry of dimension m over GF(q), i.e. PG(m, q) has number of points equal to

$$n = \frac{q^{m+1} - 1}{q - 1} \tag{5.2}$$

Any u-space $(1 \leq u \leq m-1)$ of PG(m, q) has number of points equal to

$$w = \frac{q^{u+1} - 1}{q \ 1} \tag{5.3}$$

The total number of u-spaces of PG(m, q)[11] is

$$N = \frac{\prod_{i=0}^{u} (q^{m+1} - q^{i})}{\prod_{j=0}^{u} (q^{u+1} - q^{j})} = N(u, m, q) \tag{5.4}$$

Let S be a matrix whose rows correspond to the u-space $(1 \leq u \leq m-1)$ of PG(m, q). The matrix S has N rows, n columns and w ones per row. We can regard each row of S as a code word of a zero-false-drop code. This is a constant weight code. The intersection of two u-spaces is a space of dimension u-1 or less. Hence

$$\mu_{\max} = 1 + q + \ldots + q^{u-1} \tag{5.5}$$

From equation (5.4),

$$t \geq \left[\frac{w - 1}{\mu_{\max}}\right] = \left[\frac{1 + q + \ldots + q^{u} - 1}{1 + q + \ldots + q^{u-1}}\right] = q \qquad (5.6)$$

We now show that t is equal to q. Let v_u be a code word. v_u corresponds to an u-space. Total number of (u-1)-spaces contained in this u-space orthogonal on a particular (u-2)-space is

$$\frac{(1 + q + \ldots + q^{u}) - (1 + q + \ldots + q^{u-2})}{(1 + q + \ldots + q^{u-1}) - (1 + q + \ldots + q^{u-2})} = 1 + q \qquad (5.7)$$

Let $v_u^{(i)}$ be a code word not equal to v_u and $v_u^{(i)}$ contains i-th $(1 \leq i \leq 1 + q)$ of these (u - 1)-spaces which are contained in v_u. The code word v_u is logically included in $1 + q$ different code words $v_u^{(i)}$ $(1 \leq i \leq 1 + q)$. Hence

$$t = q \qquad (5.8)$$

We obtain ZFD_q codes for any power of prime q. There exists m-1 different codes for a fixed m and q. The code lengths n for these codes are equal. Among these codes, the best one is the code with maximum number of code words N. From equation (5.4), the maximum N occurs when

$$u = \left[\frac{m}{2}\right] \qquad (5.9)$$

Similarly, we can take S to be a matrix whose rows correspond to u-dimensional flats of the Euclidean geometry of dimension m over GF(q), or EG(m,q). The number of rows N, the number of columns n, the number of ones per row will take the following values [11].

$$n = q^{m} \qquad (5.10)$$

$$N = N(u, m, q) - N(u, m-1, q) \qquad (5.11)$$

$$w = q^{u} \qquad (5.12)$$

Two u-dimensional flats EG(m, q) intersect an u'-dimensional flat ($0 \leq u' < u$) or do not intersect at all. Hence

$$\mu_{max} = q^{u-1} \tag{5.13}$$

and

$$t \geq \left[\frac{w-1}{\mu_{max}}\right] = \left[\frac{q^u - 1}{q^{u-1}}\right] = q-1 \tag{5.14}$$

Let $\alpha^{d_1}, \alpha^{d_2}, \ldots, \alpha^{d_u}$ be linearly independent over GF(q^m). Let v_u be a vector corresponding to an u-dimensional flat consists of q^u points

$$a_1 \alpha^{d_1} + a_2 \alpha^{d_2} + \ldots . + a_u \alpha^{d_u} \tag{5.15}$$

where $a_i(1 \leq i \leq u)$ runs independently over GF(q).

Let α^{d_o} be a point not in this EG(m,q). Let $v_u^{(i)}$ $(1 \leq i \leq q)$ be vectors corresponding to u-dimensional flats each of which consists of points

$$a_o \alpha^{d_o} + a_1 \alpha^{d_1} + \ldots + a_{u-1} \alpha^{d_{u-1}} + q_i q^{d_u} \tag{5.16}$$

where a_i ($0 \leq i \leq u-1$) runs independently over GF(q) and $q_1, q_2, \ldots, q_q$ are distinct elements in GF(q).

The code word v_u is logically included in the sum of these code words. Hence

$$t = q - 1 \tag{5.17}$$

There exists m-1 different ZFD_{q-1} codes for a fixed m and q. The code length n and the order of superimposed code t (t = q-1) is fixed. Among these codes, the best one is the code such that the number of code words N is the largest. We list some zero-false-drop codes derived from finite geometries in Table 5. 1.

Table 5.1 Zero-False-Drop Codes Derived From Finite Geometries

Projective Geometry Zero-False-Drop Codes

N	35	155	1395	11811	97155	130	1210	33880	357	5797	806	2850	4745
n	15	31	63	127	255	40	121	364	85	341	156	400	585
t	2	2	2	2	2	3	3	3	4	4	5	7	8

Euclidean Geometry Zero-False-Drop Codes

N	117	1080	32670	336	5440	775	2793	4672
n	27	81	243	64	256	125	343	512
t	2	2	2	3	3	4	6	7

5.3 A Method for Encoding and Retrieval of Documents

A method of encoding and retrieval of documents derived from algebraic coding theory is introduced in the first section. A new method also derived from algebraic coding theory will be presented in this section. A comparison will be made with the previous method. Let n be the code length of a t-error-correcting BCH code. Let α be the primitive n-th root of unity. If the total number of the descriptors in the dictionary is less than n, we can represent the j-th descriptors by α^{j-1}. If a document has w_d number of descriptors $X_1^{(d)}, X_2^{(d)}, \ldots, X_{w_d}^{(d)}$, we propose to represent the document the sequence of digits $\sigma = \left[\sigma_1, \sigma_2, \ldots \sigma_{w_d}\right]$ where σ_i are elemently symmetrical functions of $X_1^{(d)}, X_2^{(d)}, \ldots, X_{w_d}^{(d)}$. If a query has w_q number of descriptors $X_1^{(q)}, X_2^{(q)}, \ldots, X_{w_q}^{(q)}$, the determination of whether a document covers a query becomes to determine whether $\sigma_{w_d} - \sigma_{w_d -1} x + \ldots\ldots\ldots + \sigma (-x)^{w_d -1} + (-1)^{w_d}$ contains $X_i^{(q)}$ as roots. The hardware required to realize this decision is considerably simpler than BCH code decoder which is required for retrieval if the document is represented by the syndrome. The hardward required to encode σ is no more complicated than the hardware required to encode s_d. An information system described in reference [3] utilizes the variable length coding to minimize system requirements in both storage and computation. The new representation of document is also a variable length scheme, and the storage required is minimum. The computation seems to be simpler than to determine the weight of coset leader from the syndrome of a document plus a query.

5.4 On the Use of Finite Geometry Codes by Chien's Formulation

By using Chien's formulation [3], the main task to determine whether a document covers a query is equivalent to find the weight of coset leader from the syndrome. For the codes constructed from finite geometries, Reed Decoding Algorithm can be applied. The determination of weight of coset leader from syndrome if codes constructed from finite geometries instead of BCH codes are used. In general, the efficiency of BCH code is higher than that of finite geometry code but the difference is slight in many cases including difference-set cyclic codes [17]. We can apply these codes constructed from finite geometries for information retrieval. We first introduce Reed Decoding Algorithm and show that, in some cases, the determination of the weight of coset leader from syndrome is simpler than using Reed Decoding Algorithm.

Let L be a nxb matrix of 1's and 0's whose b columns are elements of the null space V_2 of an (n, k) binary code V_1. Any vector of the null space is essentially a parity check rule satisfied by code vectors. For any received vector v, vL is a vector of b 1's and 0's which contains 1's in the positions corresponding parity check rules that v fails to satisfy. Now consider the result of multiplying (vL) by L^T as real numbers. The result will be a vector of n components that are integers.

$$e(v) = (vL)\, L^T \tag{5.18}$$

There will be a contribution of 1 in the j-th component of e for each column of L which fails as a parity check and contain a 1 in its j-th position. Thus the j-th component of e is the number of failures of parity checks that involve the j-th symbol in the code vectors. Let $\bar{e}$ (v) be a vector whose j-th component is 1 if the j-th component e(v) exceed certain threshold and 0 otherwise, $\bar{e}$(v) is considered to be the error vector. The number of 1's in $\bar{e}$(v) is the number of errors. For a projective geometry code associated with u-spaces of PG(m, q), let L be the matrix whose columns corresponding to u-spaces. For an Euclidean geometry code associated with u-dimensional flats of EG(m, q), let L be the matrix whose columns corresponding to the u-dimensional flats with first digit deleted. If we let t equal to $\left[(q^m-1)/2\,(q^u-1)\right]$ and set threshold at r/2 where r is the number of 1's in the row of L, $\bar{e}$(v) is the error vector provided the number of errors occurred is no more than t. In information retrieval system, let the syndrome of a document plus a query be s_{d+q}, we actually compute $e(s_{d+q}) = s_{d+q}ML^T$ instead of $e(v) = vL\,L^T$ where M is a matrix such that $vL = s_{d+q}M$.

In some cases, we need to compute $s_{d+q}\,M$ only in order to distinguish the weight of coset leader from the syndrome, the computation of $s_{d+q}M$ is much simpler than the computation of $s_{d+q}\,M\,L^T$.

We demonstrate this as follows: Let L be the matrix whose columns are u-spaces of $PG(m, 2^s)$. Let v be the coset leader corresponding to s_{d+q}, then $vL = s_{d+q}M$. If for distinct weight of v, the number of 1's in $s_{d+q}M$ is distinct, the computation of $s_{d+q}M$ is sufficient to determine the weight of coset leader from the syndrome. We now show that the weight of $s_{d+q}\,M$ is distinct for s_{d+q} corresponding to the coset of weight 0, 1, 2, or 3 for t greater than or equal to three. The computation $s_{d+q}M$ is sufficient to determine the weight of coset leader from the syndrome if the code is a triple error-correcting code.

Let w(x) denote the weight of a vector x. We have

$$w(s_{d+q}M) = 0 \qquad \text{when} \qquad w(v) = 0 \tag{5.19}$$

The number of 1's in any row of L is r where

$$r = \frac{\prod_{i=(m-u+1)}^{m} (2^{is}-1)}{\prod_{j=1}^{u} (2^{js} - 1)} \tag{5.20}$$

$$w(s_{d+q}M) = r \qquad \text{when} \qquad w(v) = 1 \tag{5.21}$$

The number of pairs both with 1's in any two rows of L is λ where

$$\lambda = \frac{\prod_{i=(m-u+1)}^{m-1} (2^{is}-1)}{\prod_{j=1}^{u-1} (2^{js}-1)} \tag{5.22}$$

When weight of v is equal to two, the two nonzero digit positions correspond to points α^{d_1}, α^{d_2} of $PG(m, 2^s)$. The number of u-spaces contains any one point is r and contains any two points is λ. A parity check corresponding to an u-space fails if this u-space contains one and only one of the points α^{d_1}, α^{d_2}. The number of u-spaces containing α^{d_1} and not α^{d_2} is $r - \lambda$. Thus

$$w(s_{d+q}M) = 2(r - \lambda) \quad \text{when} \quad w(v) = 2 \tag{5.23}$$

When weight of v is equal to three, the three nonzero digit positions correspond to three points α^{d_1}, α^{d_2} and α^{d_3} of $PG(m, 2^s)$. Two cases possible. In case (1), these three points are linearly dependent. If an u-space contains two of these three points, then it must contain the third point. Total number of u-spaces containing all three points is λ. Thus

$$w(s_{d+q}M) = 3(r - \lambda) + \lambda = 3r - 2\lambda \tag{5.24}$$

In case (2), the three points are linearly independent. Let λ_1 be the number of u-spaces containing these three points.

$$\lambda_1 = \frac{\prod_{i=m-u+1}^{m-2} (2^{is} - 1)}{\prod_{j=1}^{u-2} (2^{js} - 1)} \tag{5.25}$$

The number of u-spaces containing α_1 but not α_2 and α_3 is $r - 2\lambda + \lambda_1$. The number of u-spaces containing one of the three points or all of the three points is $3(r - 2\lambda + \lambda_1) + \lambda_1$. Thus

$$w(s_{d+q}M) = 3(r - 2\lambda + \lambda_1) + \lambda_1 = 3r - 6\lambda + 4\lambda_1 \tag{5.26}$$

$$t = \left[r/2\lambda\right] \tag{5.27}$$

When t is greater than or equal to three

$$r \geq 6\lambda \tag{5.28}$$

From equation (5.28),

$$3r - 6\lambda + 4\lambda_1 > 2r - 2\lambda > r > 0 \tag{5.29}$$

also

$$3r - 2\lambda > 2r - 2\lambda \tag{5.30}$$

Thus the weight of $s_{d+q}M$ is distinct for s_{d+q} corresponding to the coset of weight 0, 1, 2, or 3.

5.5 On the Use of Symmetry of Codes for Retrieval

One form of symmetry of a systematic code is a permutation of bit positions in each code word (the same permutation is applied to all code words) which preserves the code as a whole. The idea of using symmetry of the code for information retrieval is closely related to the concept of permutation decoding [10]. The permutations which leave the code invariant have a desired property for information retrieval purposes. If G is a group of permutations that leaves the code invariant, then G partitions cosets into equivalence classes. The coset leaders of the coset in the same equivalence class have the same weight. There exists a group G_1 isomorphic to G [19]. G_1 partitions the set of syndromes into orbits. If two syndromes in the same orbit of G_1, then these corresponding cosets are in the same equivalence class of G. That two syndromes in the same orbit implies that their corresponding coset leaders must have the same weight. If a t-error-correcting code can be decoded by permutation decoding and a received sequence has errors less or equal to t, then all the errors in this received sequence can be moved to parity check portion by a permutation in G. In an equivalence class under G of cosets whose coset leaders are of weight t_1 which is less or equal to t, then there must be a coset leader with all its 1's in the parity check portion. The syndromes of the coset leader having information digit portion all zero is identical to parity check portion of this coset leader. For any syndrome corresponding coset of weight t_1 which is less or equal to t, there exists a syndrome in the same orbit under G_1 such that the number of 1's in this syndrome is t_1. It becomes clear at this point that a t-error-correcting code which can be decoded by permutation decoding can be used for information retrieval purpose. Recall that a document is represented as a syndrome in Chien's formulation and the main computation is to determine the weight of coset leader from the syndrome. The determination of whether a document covers a query is as follows. We obtain a syndrome s_{d+q} by adding the syndrome of document s_d and the syndrome of query s_q. We determine the weight of coset leader of s_{d+q} by generating the syndromes in the same orbit of s_{d+q} under G_1. If one of these syndromes has number of 1's equal to t_1 which is less or equal to t, the weight of coset leader must be equal to t_1. Otherwise the weight of coset leader is larger than t. The effective use of this principle lies on a method to generate the syndromes in the same orbit without duplication which will be presented at the end of this section. The permutation decoding is mainly for low rate codes. If a code has minimum distance d equal to 2t + 1, then this code is capable of correcting t errors. If we cannot move all errors in an error pattern with weight less than or equal to t to the parity check portion by the permutations which leave the code invariant, the permutation decoding scheme does not work for this code. If we represent the documents and queries as syndromes of this code, there exists a syndrome s_{d+q} corresponding a coset of weight less than or equal to t, but all syndromes in the orbit of s_{d+q} under G_1 has number of 1's larger than t. In this case, we need some modification in information retrieval process in order to make use of this code. Two approaches are possible. The first approach is as follows. We pick any syndrome in the orbit as a representative if syndromes in this orbit corresponding coset leaders of weight less than or equal to t and all of the syndromes have number of 1's greater than t. We store all representatives of the orbits having the preceding property and the corresponding weights of the coset leaders. Given a syndrome s_{d+q}, we proceed to generate the syndromes in the same orbit. If one of the syndromes has number of 1's less than or equal to t, we determine the weight of coset leader immediately. Otherwise we can do table look up to find its weight. In this approach, we need storage to store the table which consists of the representatives of the orbits and their corresponding weights. The approach of this method is practical provided the table is not very large. It is possible to save storage by the following second approach. We define a vector v_c covers a coset leader e if the information portion of v_c agrees with any coset leader e' in the equivalence class of e under G. We obtain a set of covering vectors that cover every coset leader. We call their corresponding syndromes the covering syndromes. For any syndrome of a document plus a query, we can find a syndrome in the same orbit and the sum of this syndrome and a covering syndrome will have the number of 1's less than or equal to t in the resulting syndrome. In this case, we are able to determine whether a particular document covers a given query. The retrieval process is based on this principle. In the second approach, one covering syndrome may cover several equivalence classes. The storage requirement in the second approach is less than that in the first approach. In the first approach, we can arrange syndromes having the same weight together in the table. We need only to check whether the syndrome in the equivalence class of s_{d+q} match the syndromes in the section of the table where syndromes have weight equal to the weight of document minus the weight of query. In the second approach, we can arrange the covering syndromes with same weight together in the table, we need only to check whether the syndromes in the equivalence class of s_{d+q} are covered by the syndromes which have weight no more than the weight of document minus the weight of query.

A method to generate syndromes in the same orbit without duplication is as follows. The only known group of permutations which leaves any binary cyclic code invariant is the group Gn generated by cyclic permutation T and the permutation U which maps ω to 2ω mod n where ω is coordinate number labeled as $0, 1, 2, \ldots, n-1$ [10]. If n is odded, there exists a least integer t such that $2^t \equiv 1 \bmod n$ and $U^t = I$. It is easy to check that $TU = UT^2$ (i.e. $vTU = vUT^2$ for any vector v), hence we may represent every permutation in Gn in the form U^iT^j with $0 \leq i \leq t-1$, $0 \leq j \leq n-1$. Now every power of U leaves 0 fixed; thus $U^iT^j = U^hT^k$ if and only if $i = h \bmod t$ and $j = k \bmod n$. Thus the group Gn consists of nt permutations U^iT^j for all i, j such that $0 \leq i \leq t-1$ and $0 \leq j \leq n-1$.

Gn partitions cosets into equivalence classes. The weight preserving group which is isomorphis to Gn partitions the set of syndromes into orbits. We want to generate the syndrome in the orbit without duplication. Otherwise it is wasting time. This [illegible] same problem to get all coset leaders in an equivalence class without [illegible]ion. Let v be a coset leader. If the equivalence class of v consists of nt distinct elements, then all the elements in the following t x n matrix are the distinct coset leaders in the equivalence class.

$$\begin{bmatrix} vU^iT^j \end{bmatrix}_{t \times n} \qquad (5.31)$$

where $0 \le i \le t-1, \quad 0 \le j \le n-1.$

If the equivalence class of v does not have nt distinct elements, the distinct elements can be obtained as follows.

Let e be the smallest positive integer such that $vT^e = vI$ and t' be the smallest positive integer such that $vU^{t'} = vT^j$ for some j. The equivalence class of v containing the elements

$$vU^iT^j \qquad \text{for} \quad 0 \le i \le t'-1, \quad 0 \le j \le e-1 \qquad (5.32)$$

The proof is as follows. The elements in any row are just cyclic shift of each other. If t' is the smallest integer such that $vU^{t'} = vT^j$ for some j. The elements in t'+ 1-th row of matrix (5.31) are identical to the elements in the first row of matrix (5.31). The elements in the t' + 2 row is identical to the elements in the second row etc. If the row consists of vU^{i_1} is identical to the row consists of vU^{i_2} where $0 \le i_1 < i_2 \le t'-1$, then the row consists of $vU^{i_2-i_1}$ is identical to the first row. This is a contradiction because $i_2 - i_1 < t'$. Thus the first t' rows of matrix (5.31) are the only distinct rows. We define the period of a vector v to be the smallest integer e such that $vT^e = vI$. We need to show that the period of v is equal to the period of vU^i.

Let v(x) be a polynomial whose coefficients corresponding to v , and the period of v be e. Let $v(x^2)$ be a polynomial whose coefficients corresponding to vU and the period of vU be e'.

$$x^e v(x) = v(x) \qquad \text{mod} \;\; x^n - 1 \qquad (5.33)$$

implies that

$$x^e v(x^2) = x^e (v(x))^2 = (x^e v(x))\, v(x) = v(x)\, v(x) = v(x^2) \;\text{mod}\; x^{n-1} \quad (5.34)$$

Therefore

$$e' \leq e \qquad (5.35)$$

$$x^{e'} v(x^2) = v(x^2) \quad \text{mode} \;\; x^n - 1 \qquad (5.36)$$

implies that

$$x^{e'} v(x) = x^{e'} (v(x^{2^t})) = x^{e'} (v(x))^{2^t} = x^{e'} (v(x))^2 \, (v(x))^{2^t - 2}$$

$$= x^{e'} v(x^2) \, (v(x))^{2^t - 2} = v(x^2) \, (v(x))^{2^t - 2} = v(x) \quad \text{mod} \; x^n - 1 \qquad (5.37)$$

Therefore

$$e' \geq e \qquad (5.38)$$

From equations (5.35) and (5.38)

$$e' = e \qquad (5.39)$$

We have shown that $e' = e$ for $i = 1$. Similar argument will enable us to prove that vU^i and $(vU^i)U$ have same period, hence v and vU^i have same period for any i.

Thus we have shown that the equivalence class of v containing the elements defined in equation (5.32).

5.6 Investigation of Using Concatenated Codes

In Chien's formulation, we represent the descriptors by the columns of the parity check matrix of a t-error-correcting code where t is the maximum number of descriptors allowed by any document. It is easy to verify that a code V_1' obtained by joining severial t-error-correcting codes is also capable of correcting t-errors, therefore we can represent the descriptors by the parity check matrix of V_1'. The use of multiple copies of a single code will simplify the computational process but on the other hand, will require larger addresses. The situation is roughly as follows.

If one BCH code with length equal to $2^m - 1$ is used, the number of parity check digit is mt. Now if we use $2^{m'}$ number of the same code with code length $2^{m-m'}-1$. The number of allowable descriptors in the dictionary is $2^{m'}(2^{m-m'}-1)$ which is a little less than $2^m - 1$ when m is much great. that m'. Let ℓ equal to $2^{m'}$. We will show that the fractional amount of work required by using ℓ codes compared with that required by using one code for each different weight of documents is $\ell^{-2D}\sum_k C_k^2$ where D is the weight of the document and the C_k's are the coefficients of the terms in the expansion of $(x_1 + x_2 + \ldots + x_\ell)^D$. To illustrate, for D = 10, the fractional amount of work by using four copies is 0.01. The detail derivation of the formula and a curve for showing the percentage amount of work required by using ℓ BCH codes as compared with that of one BCH code for different weights of a document is as follows.

Let D be the weight of a document and let Q be the weight of a query. If D is greater than Q, we can determine whether the document covers the query by comparing the syndromes corresponding to the cosets of weight D-Q. Let A_i be the number of cosets of weight i, then $A_i = \binom{n}{i}$ where n is the code length. The number of comparison is A_i. We can use a concatenated code obtained by joining two codes together. Given a query of weight Q with weight Q_1 in the first section and weight $Q-Q_1$ in the second section. We do not need to test all the documents of weight D which is greater than Q. We need only to tes' the documents of weight D and with weight D_1 in the first section where $Q_1 \leq D_1 \leq D-Q+Q_1$. For a document of weight D with weight D_1 on the first section, we need to make rouhly $\binom{D-Q}{D_1-Q_1} A_{D-Q}/2^{D-Q}$ comparisons to determine whether a document covers a query. We assume the probability of occurrence of each descriptor to be equal. The percentage of documents of weight D_1 on the first section and $D-D_1$ on the second section in all documents of weight D is $\binom{D}{D_1}/2^D$. Let θ_2, θ_1 be the total amount of work for testing all documents of weight D by using two, one codes respectively. Then

$$\frac{\theta_2}{\theta_1} = \frac{\sum\limits_{Q_1 \leq D_1 \leq D-Q+Q_1} \binom{D-Q}{D_1-Q_1}\binom{D}{D_1}}{2^{D-Q}\ 2^{D}} = \frac{\sum\limits_{i=0}^{D-Q} \binom{D-Q}{i}\binom{D}{i+Q_1}}{2^{2D-Q}} \tag{5.40}$$

The probability of a query with weight Q_1 in the first section and $Q-Q_1$ in the second section is approximately $\binom{Q}{Q_1}/2^Q$. On the average the percentage amount of work for a query of weight Q is

$$\sum_{Q_1=0}^{Q} \frac{\binom{Q}{Q_1}}{2^Q}\left[\frac{\sum\limits_{i=0}^{D-Q}\binom{D-Q}{i}\binom{D}{i+Q_1}}{2^{2D-Q}}\right] = \frac{\sum\limits_{i=0}^{D}\binom{D}{i}^2}{2^{2D}} = 2^{-2D}\sum_k C_k^2 \tag{5.41}$$

where C_k's are the coefficients of the terms in the expansion of $(x_1 + x_2)^D$. Generally, if we use a concatenated code obtained by joining ℓ BCH codes together. The fractional amount of work required by using a concatenated code and a BCH code is $\ell^{-2D}\sum_k C_k^2$ where D is the weight of the document and the C'_ks are the coefficients of the terms in the expansion of $(x_1 + x_2 + \ldots + x_\ell)^D$. The derivation of this formular is similar to the derivation of the formula for $\ell = 2$ case. The percentage amount of work using ℓ BCH codes comparing with one BCH code under different weight of documents is plotted in Fig. 1. The curves are independent of the weight of a query.

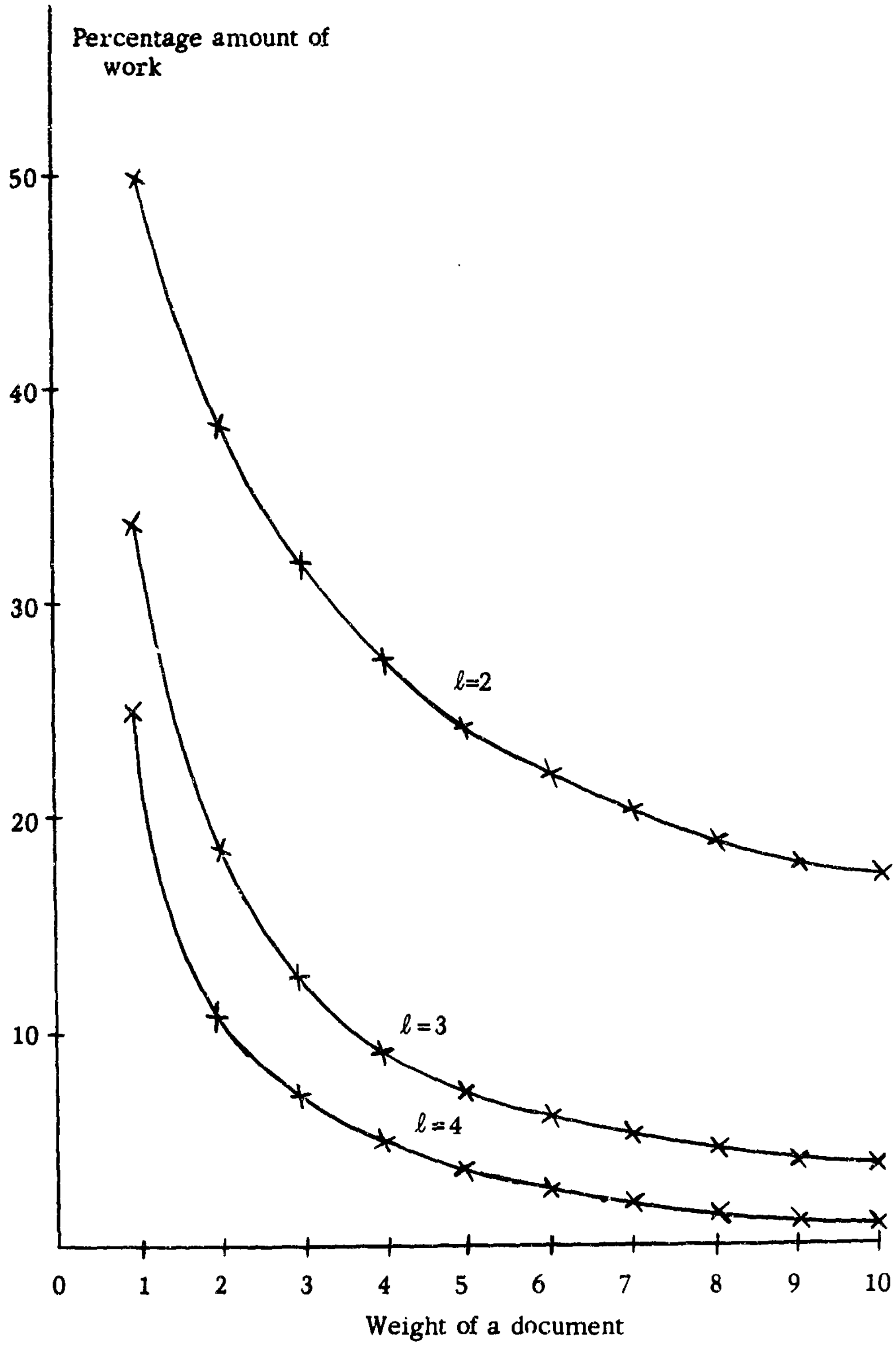

Fig. 1. The percentage amount of work required by using ℓ BCH codes as compared with that of one BCH code for different weights of a document.

VI. CONCLUSIONS AND FURTHER PROBLEMS

6.1 Conclusions

Two related classes of codes derived from Euclidean geometries have been found. These codes can be u-step decoded by threshold decoding. These codes are comparable to projective geometry codes which are moderately efficient random-error-correcting codes for practical values of code length and rate and can be decoded with a relatively modest amount of equipment. Hence it appears that these codes may be suitable for use in error control systems requiring random-error correction. The polynomial version of Rudolph's projective geometry codes has been found for practial values of code length and rate. It is important to find the polynomial version of a cyclic code because we need to know the generator polynomial for encoding purposes and furthermore the number of information digits can be determined easily from the generator polynomial. One-step threshold decoding using not necessarily orthogonal parity checks is found to be applicable to BCH codes and codes derived from Euclidean geometries. We found that it is possible to improve upon Rudolph's decoding methed in some cases. If projective geometry codes, Euclidean geometry codes, and BCH codes are decoded by one-step threshold decoding methods, their efficiency and error-correcting ability are comparable. Two new classes of zero-false-drop codes have been found which compare favorably with the previously known classes. Several results related to the application of algebraic coding theory are obtained. They may be useful in a practical information retrieval system.

6.2 Further Research Areas

Because of the easy implementation of threshold decoder and abundant results on the construction of threshold-decodable codes, it appears that threshold decoding will continue to be a promising area of research directed toward improving the reliability of data transmission in communication systems. Some promising areas for future research are:

1. To find more powerful threshold decoding algorithms and evaluate error-correcting ability of these algorithms for cyclic codes.

We have shown that not all cyclic codes can be L-step orthogonalized. The L-step orthogonalization procedure can be generalized by allowing the use of non-orthogonal parity checks at each step. One-step decoding methods are the only ones that have been investigated by means of their relation to balanced incomplete block designs. As is well known, many error patterns with weights greater that $\left[(d-1)/2\right]$, where d is the minimum distance of the code, can be corrected by threshold decoding. It would be desirable to find more powerful threshold decoding algorithms and to evaluate more precisely error-correcting ability of threshold decoding algorithms.

2. To construct new codes suitable for threshold decoding or to improve the threshold decoders for known codes.

The finite geometry codes are not as numerous as BCH codes. It is easy to define a class of codes which contains the Euclidean geometry codes and projective geometry codes as subclasses, as follows.

Let α be a primitive element of $GF(q^m)$. Let a be an integer which divides q^m-1. Let C be a code whose generator polynomial contains α^h as roots for h which are less than q^m-1, are divisible by a, and satisfy the condition $0 \leq w_q(hp^j) \leq I$ where $0 \leq j \leq s-1$ and I is a fixed integer. When I is a multiple of (q-1) and a is equal to one, we have a Euclidean geometry code. When I is a multiple of q-1 and a is equal to q-1, we have a projective geometry code. For other values of I and a we obtain codes that have not been investigated previously. The number of information digits and the BCH bound for these codes can be determined easily. To establish a decoding algorithm for these codes, possible by exploiting their geometrical properties, would be very useful.

Improvement of the decoders for finite geometry codes are possible. The choice of u'-dimensional flat parity checks in the implementation of u-step decoders is not unique. It is quite possible that one choice would lead to simpler circuitry than the others. The number of majority gates may be reduced by detailed evaluation of the dependency of these u'-dimensional flat parity checks. As to one-step decoders for projective geometry codes, we have demonstrated in section 3.4 the possibility of using some but not all u-dimensional flats for majority voting and still achieving the same guaranteed decodable distance as Rudolph's method does. The number of parity checks in the one-step decoding algorithm for Euclidean geometry codes and BCH codes proposed in section 4.2 may be more than enough. The guaranteed decodable distance $1 + \lfloor (q^m-1) / (q^u-1) \rfloor$ for one-step decoding of finite geometry codes by this decoder is only a bound which may underestimates the error-correcting ability of the decoder. Further investigation of these questions may lead to fruitful results.

LIST OF REFERENCES

1. Berman, G., "Finite Projective Geometries," The Canadian Journal of Mathematics, Vol. IV, No. 3, 1952.

2. Carmichael, R.D., Introduction to the Theory of Groups of Finite Order, Dover Publication, Inc., 1937.

3. Chien, R.T., and Frazer, W.D., "Application of Coding Theory to Document Retrieval," Presented at 1966 IEEE International Symposium on Information Theory, Los Angeles, California, 1966.

4. Graham, R.L. and MacWilliams, J., "On the Number of Information Symbols on Difference-Set Cyclic Codes," B.S.T.J. vol. XLV, no. 7, pp. 1057-1071, Sept. 1966.

5. Kasami, T., "A Decoding Procedure For Multiple-Error-Correcting Cyclic Codes," IEEE Trans., IT-10, no. 2, pp. 134-139, April, 1964.

6. Kasami, T. et al, "Some Results on Cyclic Codes which are Invariant under the Affine Group," AFCRL-66-622, Aug., 1966.

7. Kasami, T. and Lin, S., "Some Codes which are Invariant under Doubly Transitive Permutation Group and Their Connection with Balanced Incomplete Block Design," AFCRL-66-142, Jan., 1966.

8. Kasami, T. "Weight Distribution Formula for Some Cyclic of Cyclic Codes," Report R-285, Coordinated University of Illinois, April, 1966.

9. Kautz, W.H. and Singleton, R.C., "Nonrandom Binary Superimposed Codes," IEEE Trans., IT-10, no. 4, pp. 363-378, Oct., 1964.

10. MacWilliams, J., "Permutation Decoding of Systematic Codes," B.S.T.J., vol. XLIII, no. 1, part 2, pp. 485-505., Jan., 1964.

11. Mann, H.B., Analysis and Design of Experiments, Dover Publications, Inc., 1949.

12. Massey, J., "Advances in Threshold Decoding," to appear in Advances in Communication Systems, Vol. 2, Edited by A. V. Balakrishuan, to be published by Academic Press, New York.

13. Massey, J., Threshold Decoding. The MIT Press, 1963.

14. Peterson, W.W., Error-Correcting Codes, The MIT Press, 1961.

15. Peterson, W.W. "On the Weight Structure and Symmetry of BCH Codes," AFCRL-65-515, July, 1965.

16. Rudolph, L. D., "A Class of Majority Logic Decodable Codes," IEEE Trans., IT-13, no. 2, pp. 305-306, April, 1967.

17. Weldon, E.J. Jr., "Difference-Set Cyclic Codes," B.S.T.J. vol. XLV, no. 7, pp. 1045-1057, Sept., 1966.

18. Weldon, E.J. Jr., "Non-primitive Reed-Muller Codes," Presented at First Princeton Symposium on Circuit Theory, March, 1967.

19. Zieler, N., "On Decoding Linear Error-Correcting Codes--I," IRE Trans., IT-6, pp. 450-459.

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*

University of Illinois
Coordinated Science Laboratory
Urbana, Illinois 61801

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

A GEOMETRIC APPROACH TO CODING THEORY WITH APPLICATION TO INFORMATION RETRIEVAL

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*

Chow, David K.

6. REPORT DATE October, 1967	7a. TOTAL NO. OF PAGES 81	7b. NO. OF REFS 19
8a. CONTRACT OR GRANT NO. DAAB-07067-C-0199;Also in part NSF GK-690 b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) R-368 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*	

10. DISTRIBUTION STATEMENT

Distribution of this report is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Joint Services Electronics Program thru U.S. Army Electronics Command Ft. Monmouth, New Jersey 07703

13. ABSTRACT

Finding cyclic codes that can be decoded efficiently by threshold logic is important because the decoders are very easy to implement. Two related classes of codes derived from Euclidean geometries are presented. The code length, number of information symbols, and minimum distance are shown to be related by means of parameters of a code. These codes can be decoded with a variation of the original algorithm proposed by Reed for Reed-Muller codes. We show that these codes are comparable to Rudolph's projective geometry codes which are known to have the following important feature. For a given code length and rate, the projective geometry code has relatively large minimum distance and the decoder is usually very simple. We have dereived a class of codes from projective geometries in terms of the roots of generator polynomials. These codes are shown to contain the corresponding non-primitive Reed-Muller codes discovered by Weldon (18) as subcodes, in many cases, proper subcodes with the same error-correcting ability by L-step orthogonalization procedure. These codes are found to be identical to Rudolph's projective geometry codes for all useful parameters of the codes. Threshold decoding of BCH codes and the generality of L-step orthogonalization procedure to cyclic codes are discussed. Investigation on the application of coding theory to information retrieval is presented.

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
FINITE GEOMETRY CODES						
THRESHOLD DECODING						
INFORMATION RETRIEVAL						